JFROG eclipse milo security issues

Parent Dependency: org.eclipse.milo:sdk-core:jar:0.6.3:compile
Version used: 0.6.3

<**dependency**>
	<**groupId**>org.eclipse.milo</**groupId**>
	<**artifactId**>sdk-core</**artifactId**>
	<**version**>0.6.3</**version**>
</**dependency**>

Summary	Severity	Component	Infected Version	Fix Version
Netty codec/ src/main/ java/io/netty/ handler/ codec/ compr ession/ Lz4FrameEn coder.java Lz4FrameEn coder::finish Encode() Function Buffer Overflow	Critical	io.netty:nett y-codec	< 4.1.66.Final	4.1.66.Final

The Snappy frame decoder function doesn't restrict the chunk length which may lead to excessive memory usage. Beside this it also may buffer reserved skippable chunks until the whole chunk was received which may lead to excessive memory usage as well. This vulnerability can be triggered by supplying malicious input that decompresses to a very big size (via a network stream or a file) or by sending a huge skippable chunk.	High	io.netty:netty-codec	< 4.1.68.Final	4.1.68.Final

The Bzip2 decompression decoder function doesn't allow setting size restrictions on the decompressed output data (which affects the allocation size used during decompression). All users of Bzip2Decoder are affected. The malicious input can trigger an OOME and so a DoS attack	High	Io.netty:netty-codec	< 4.1.68.Final	4.1.68.Final
Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol	Medium	io.netty:netty-codec-http	< 4.1.71.Final	4.1.71.Final

servers & clients. Netty prior to version 4.1.7.1.Final skips control chars when they are present at the beginning / end of the header name. It should instead fail fast as these are not allowed by the spec and could lead to HTTP request smuggling. Failing to do the validation might cause netty to "sanitize" header names before it forward these to another remote system when used as proxy. This remote system can't see the invalid usage anymore,					

and therefore does not do the validation itself. Users should upgrade to version 4.1.7.1.Final to receive a patch.				
Netty is an open-source, asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers & clients. In Netty before version 4.1.59.Final there is a vulnerability on Unix-like systems involving an insecure temp file. When netty's multipart	Medium	io.netty:netty-handler	< 4.1.59.Final	4.1.59.Final

decoders are used local information disclosure can occur via the local system temporary directory if temporary storing uploads on the disk is enabled. On unix-like systems, the temporary directory is shared between all user. As such, writing to this directory using APIs that do not explicitly set the file/ directory permissions can lead to information disclosure. Of note, this does not impact modern MacOS Operating Systems. The method "File.createTempFile" on unix-like					

systems creates a random file, but, by default will create this file with the permissions "-rw-r--r--". Thus, if sensitive information is written to this file, other local users can read this information. This is the case in netty's "AbstractDiskHttpData" is vulnerable. This has been fixed in version 4.1.59.Final. As a workaround, one may specify your own "java.io.tmpdir" when you start the JVM or use "DefaultHttpDataFactory.setBaseDir(...)" to set the directory to something that is only					

readable by the current user.				
Netty is an open-source, asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers & clients. In Netty before version 4.1.59.Final there is a vulnerability on Unix-like systems involving an insecure temp file. When netty's multipart decoders are used local information disclosure can occur via the local system temporary	Medium	io.netty:netty-common	< 4.1.59.Final	4.1.59.Final

directory if temporary storing uploads on the disk is enabled. On unix-like systems, the temporary directory is shared between all user. As such, writing to this directory using APIs that do not explicitly set the file/ directory permissions can lead to information disclosure. Of note, this does not impact modern MacOS Operating Systems. The method "File.createTempFile" on unix-like systems creates a random file, but, by default will create this file with the permissions "-rw-r--r--".					

Thus, if sensitive information is written to this file, other local users can read this information. This is the case in netty's "AbstractDiskHttpData" is vulnerable. This has been fixed in version 4.1.59.Final. As a workaround, one may specify your own "java.io.tmpdir" when you start the JVM or use "DefaultHttpDataFactory.setBaseDir(...)" to set the directory to something that is only readable by the current user.				
Netty is an open-source, asynchronou	Medium	io.netty:netty-codec-http	< 4.1.59.Final	4.1.59.Final

s event-driven network application framework for rapid development of maintainable high performance protocol servers & clients. In Netty before version 4.1.59.Final there is a vulnerability on Unix-like systems involving an insecure temp file. When netty's multipart decoders are used local information disclosure can occur via the local system temporary directory if temporary storing uploads on the disk is enabled. On unix-like systems, the temporary					

directory is shared between all user. As such, writing to this directory using APIs that do not explicitly set the file/ directory permissions can lead to information disclosure. Of note, this does not impact modern MacOS Operating Systems. The method "File.createTempFile" on unix-like systems creates a random file, but, by default will create this file with the permissions "-rw-r--r--". Thus, if sensitive information is written to this file, other local users can read this information.					

This is the case in netty's "AbstractDiskHttpData" is vulnerable. This has been fixed in version 4.1.59.Final. As a workaround, one may specify your own "java.io.tmpdir" when you start the JVM or use "DefaultHttpDataFactory.setBaseDir(...)" to set the directory to something that is only readable by the current user.				

Netty is an open-source, asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers & clients. In Netty (io.netty:netty-codec-http2) before version 4.1.60.Final there is a vulnerability that enables request smuggling. If a Content-Length header is present in the original HTTP/2 request, the field is not validated by `Http2MultiplexHandler` as it is propagated up. This is fine as long as the request is	Medium	io.netty:netty-codec-http	< 4.1.60.Final	4.1.60.Final