```
################################################################
#         Default Networking Configuration File
#
# This file may contain default values for the networking system properties.
# These values are only used when the system properties are not specified
# on the command line or set programmatically.
# For now, only the various proxy settings can be configured here.
################################################################


# Whether or not the DefaultProxySelector will default to System Proxy
# settings when they do exist.
# Set it to 'true' to enable this feature and check for platform
# specific proxy settings
# Note that the system properties that do explicitly set proxies
# (like http.proxyHost) do take precedence over the system settings
# even if java.net.useSystemProxies is set to true.

java.net.useSystemProxies=false

#-----------------------------------------------------------------------
# Proxy configuration for the various protocol handlers.
# DO NOT uncomment these lines if you have set java.net.useSystemProxies
# to true as the protocol specific properties will take precedence over
# system settings.
#-----------------------------------------------------------------------


# HTTP Proxy settings. proxyHost is the name of the proxy server
# (e.g. proxy.mydomain.com), proxyPort is the port number to use (default
# value is 80) and nonProxyHosts is a '|' separated list of hostnames which
# should be accessed directly, ignoring the proxy server (default value is
# localhost & 127.0.0.1).
#
 http.proxyHost=win-proxy.services.datevnet.de
 http.proxyPort=8880
 http.nonProxyHosts=localhost|127.*|[::1]
#
# HTTPS Proxy Settings. proxyHost is the name of the proxy server
# (e.g. proxy.mydomain.com), proxyPort is the port number to use (default
# value is 443). The HTTPS protocol handlers uses the http nonProxyHosts list.
#
 https.proxyHost=win-proxy.services.datevnet.de
 https.proxyPort=8880
#
# FTP Proxy settings. proxyHost is the name of the proxy server
# (e.g. proxy.mydomain.com), proxyPort is the port number to use (default
# value is 80) and nonProxyHosts is a '|' separated list of hostnames which
# should be accessed directly, ignoring the proxy server (default value is
# localhost & 127.0.0.1).
#
 ftp.proxyHost=win-proxy.services.datevnet.de
 ftp.proxyPort=8880
 ftp.nonProxyHosts=localhost|127.*|[::1]
```

```
#
# Gopher Proxy settings. proxyHost is the name of the proxy server
# (e.g. proxy.mydomain.com), proxyPort is the port number to use (default
# value is 80)
#
# gopher.proxyHost=
# gopher.proxyPort=80
#
# Socks proxy settings. socksProxyHost is the name of the proxy server
# (e.g. socks.domain.com), socksProxyPort is the port number to use
# (default value is 1080)
#
 socksProxyHost=
 socksProxyPort=1080
#
# HTTP Keep Alive settings. remainingData is the maximum amount of data
# in kilobytes that will be cleaned off the underlying socket so that it
# can be reused (default value is 512K), queuedConnections is the maximum
# number of Keep Alive connections to be on the queue for clean up (default
# value is 10).
# http.KeepAlive.remainingData=512
# http.KeepAlive.queuedConnections=10


# Authentication Scheme restrictions for HTTP and HTTPS.
#
# In some environments certain authentication schemes may be undesirable
# when proxying HTTP or HTTPS.  For example, "Basic" results in effectively the
# cleartext transmission of the user's password over the physical network.
# This section describes the mechanism for disabling authentication schemes
# based on the scheme name. Disabled schemes will be treated as if they are not
# supported by the implementation.
#
# The 'jdk.http.auth.tunneling.disabledSchemes' property lists the
authentication
# schemes that will be disabled when tunneling HTTPS over a proxy, HTTP CONNECT.
# The 'jdk.http.auth.proxying.disabledSchemes' property lists the authentication
# schemes that will be disabled when proxying HTTP.
#
# In both cases the property is a comma-separated list of, case-insensitive,
# authentication scheme names, as defined by their relevant RFCs. An
# implementation may, but is not required to, support common schemes whose names
# include: 'Basic', 'Digest', 'NTLM', 'Kerberos', 'Negotiate'.  A scheme that
# is not known, or not supported, by the implementation is ignored.
#
# Note: This property is currently used by the JDK Reference implementation. It
# is not guaranteed to be examined and used by other implementations.
#
#jdk.http.auth.proxying.disabledSchemes=
jdk.http.auth.tunneling.disabledSchemes=Basic


#
# Transparent NTLM HTTP authentication mode on Windows. Transparent
```

```
authentication
# can be used for the NTLM scheme, where the security credentials based on the
# currently logged in user's name and password can be obtained directly from the
# operating system, without prompting the user. This property has three possible
# values which regulate the behavior as shown below. Other unrecognized values
# are handled the same as 'disabled'. Note, that NTLM is not considered to be a
# strongly secure authentication scheme and care should be taken before enabling
# this mechanism.
#
# Transparent authentication never used.
#jdk.http.ntlm.transparentAuth=disabled
#
# Enabled for all hosts.
#jdk.http.ntlm.transparentAuth=allHosts
#
# Enabled for hosts that are trusted in Windows Internet settings
#jdk.http.ntlm.transparentAuth=trustedHosts
#
jdk.http.ntlm.transparentAuth=disabled
```