Proxy

In case you have any trouble with proxy settings (for example old Cisco proxy), that information would be helpful:

on Linux:
Linux

on Windows:
Windows

On macOS:
macOS

Advanced Mode for Developers (Windows only)

Manual proxy configuration

Following article contains short information about what is proxy, and fixes for common issues.

The entire article, with more information on proxy issues with applications other than Azure DevOps, or iboss proxy information in general, can be found at Yammer.

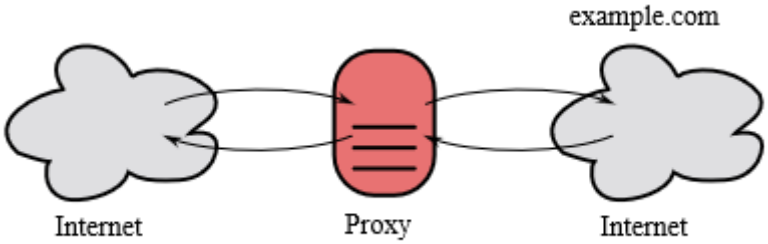
For further details InfoSec Network Security Management **CH-infosec-network-security-mgmt@abb.com** can be contacted.

Content

1 WHAT IS PROXY?

In computer networking, a proxy server is a server application or appliance that acts as an intermediary for requests from clients seeking resources from servers that provide those resources. A proxy server thus functions on behalf of the client when requesting service, potentially masking the true origin of the request to the resource server. A proxy server may reside on the user's local

computer, or at any point between the user's computer and destination servers on the Internet. A proxy server that passes unmodified requests and responses is usually called a gateway or sometimes a tunneling proxy.



2 GENERAL INFORMATION

ABB changes its Secure Web Gateway service to iboss solution.

By default, iboss service relies on Proxy Auto Configuration file enforced on the client. There are multiple ways to deliver proxy information to clients such as:

– iboss Cloud Connector
– PAC configuration in browser

In cases where none of the methods above work it is also possible to manually configure proxy gateways URLs or configure site-to-site tunnel to iboss gateway providing transparent proxy service.

3 INFRASTRUCTURE REQUIREMENTS

ABB environment must be prepared in advance to enable iboss proxy service. The following elements are required

3.1 ACTIVE DIRECTORY

Active Directory groups are required to grant access to iboss service. All the groups must be synchronized with Azure Active Directory.

3.2 CLIENTS

iboss service requires installation of iboss Connector. This is a software agent responsible for client registration and transparent authentication within iboss service.

3.3 FIREWALLS

iboss requires several ports to be opened on firewalls for a service being functional. The following ports need to be opened:

80 TCP – Proxy Autoconfiguration Configuration file location
443 TCP – client registration port
8016 TCP – certificate distribution port
8082 TCP – proxy gateway port

All the ports must be opened outbound to iboss IP ranges.

3.4 SCCM

SCCM is required to perform initial installation and further updates on ABB Windows clients.

3.5 GPO

Any changes in iboss configuration must be distributed to Windows registry over GPO. This includes iboss certificate delivery to all Windows systems

4 IBOSS CLIENT DEPLOYMENT

4.1 WINDOWS

By default, standard ABB Windows clients receive iboss installation via SCCM.
In case the automated distribution is not possible, there are installation packages available.
Installation packages contain ABB specific settings. Administrative rights are required for installation.
iboss connector runs as a service on Windows. It has two main roles – registering a client to iboss service and enforcing OS-wide proxy settings:

If the settings are altered by user, iboss connector restores them almost in real-time.
Internet Explorer, Edge, Edge-Chromium and Chrome browsers leverage OS-wide proxy settings. In case of Firefox browser, the iboss service closes it without warning to have the proxy settings being effectively enforced inside this application.

4.2 MACOS

By default, macOS clients receive iboss installation via MDM service.
In case the automated distribution is not possible, there are installation packages available.
Installation packages contain ABB specific settings. Administrative rights are required for installation.

The iboss connector at start registers a client in iboss service. It also downloads iboss certificate if needed.

iboss connector launches local proxy available at http://127.0.0.1:8009. The connector enforces the proxy settings in OS settings. Applications may use proxy service without explicit authentication.

4.3 LINUX

iboss is currently supported on the following Linux platforms: RedHat, CentOS, Debian, Ubuntu and openSUSE.
Installation packages are available on demand. Installation packages contain ABB specific settings.
“root” rights are required for installation.

The iboss connector (a service called ibsa) at start registers a client in iboss cloud proxy service. It also downloads iboss certificate if needed. The certificate is added to OS certification store:

/etc/pki/ca-trust/extracted/openssl/ca-bundle.trust.crt

iboss connector launches local proxy available at http://127.0.0.1:8009. There are no OS-wide proxy settings on Linux. De-facto standard are environmental variables and they are configured by iboss as follows:

- http_proxy=http://127.0.0.1:8009
- https_proxy=http://127.0.0.1:8009
- ftp_proxy=http://127.0.0.1:8009
- no_proxy=127.0.0.1,localhost,.abb.com

Please note that in case modification of “no_proxy” variable is needed, it cannot contain subnets inside (such as 10.1.1.0/24). Only IP addresses and domain suffixes are supported, however there are tools that interpret this variable in a different way. Please note as well that some tools may require the variables to be defined in upper or lower case. The “no_proxy” variable should not be directly customized. Instead the setting “IPBypass” in /usr/share/ibsa/config.json should be populated with required entries (which gets effective after ibsa service restart). E.g.:
"IPBypass": "127.0.0.1,localhost,codebits.abb.com,tfsa.abb.com",
All applications that support environmental variables mentioned above may leverage iboss service without explicit authentication.
Browsers on Linux platforms should use PAC file which is accessible at the following URL:
http://127.0.0.1:8000/proxy.pac

5 ADVANCED MODE FOR DEVELOPERS (WINDOWS ONLY)

Advanced configuration is available only on Windows platform. In this configuration iboss Connector works in so called “HTTP Tunnel” mode. In this mode all outbound TCP connections are transparently tunneled from a client to iboss inspection gateways. This feature covers in addition the following cases:

- Applications or tools that do not support proxy configuration
- Applications or tools that using non-Web protocols (such as SSH, RDP, etc.)

UDP and ICMP traffic cannot be tunneled the same way.

5.1 INSTALLATION ADVANCED

Installations packages can be provided on-demand but on standard ABB clients iboss Connector installation will be automatically performed by SCCM. In case the automated distribution is not possible, there are installation packages available.

5.2 CONFIGURATION

Proper configuration of iboss connector for HTTP Tunnel mode is done automatically and is based on AD group membership. Each developer requiring HTTP Tunnel mode to be enabled must request !ZH_InternetAccessDev AD group membership. For the time being such request should be submitted by mail to InfoSec Network Security Management CH-infosec-network-security-mgmt@abb.com team.

The information below has been provided for additional configuration steps which are neccessery. In the future it won't be needed to configure anything manually.
HKEY_LOCAL_MACHINE\SOFTWARE\Iboss\IBSA\Parameters

To enable HTTP Tunnel mode the following entries must be set:
"RedirectAllPorts"=dword:00000001
"UseHttpTunneling"=dword:00000001

The following additional settings should be configured in addition, to provide seamless operation in ABB environment.

Traffic to local ABB IP subnets and Office 365 IP ranges should bypass proxy tunneling. Other networks can be added if required.
"IPBypassList"="138.221.0.0/16,138.222.0.0/15,138.224.0.0/16,138.226.0.0/15,138.228.0.0/16,130.110.0.0/16,137.133.128.0/17,145.241.0.0/16,193.164.16.0/20,192.125.60.0/22,193.45.16.0/20,193.71.72.0/24,193.75.73.0/24,198.113.64.0/18,129.35.205.0/24,129.35.207.0/24,80.75.192.0/26,80.75.195.0/26,204.231.112.0/20,206.208.208.0/23,210.24.147.0/24,13.107.6.152/31,13.107.18.10/31,13.107.128.0/22,23.103.160.0/20,40.96.0.0/13,40.104.0.0/15,52.96.0.0/14,131.253.33.215/32,132.245.0.0/16,150.171.32.0/22,191.234.140.0/22,204.79.197.215/32,40.92.0.0/15,40.107.0.0/16,52.100.0.0/14,104.47.0.0/17,13.107.64.0/18,52.112.0.0/14,13.107.136.0/22,40.108.128.0/17,52.104.0.0/14,104.146.128.0/17,150.171.40.0/22"

The following applications should not be tunneled transparently. They should use standard proxy settings. Other applications may be added if required. Process names must be provided in case sensitive format.
"TransparentTunnelAppExList"="chrome.exe,firefox.exe,iexplore.exe,msedge.exe,MicrosoftEdge.exe,OUTLOOK.EXE,Teams.exe,lync.exe,vpnui.exe,vpnagent.exe,vpndownloader.exe,CiscoCollabHost.exe,RingCentralMeetings.exe,zoom.exe"

It is possible to configure only selected TCP ports to be tunneled. For example, to allow only MS SQL, SSH and Remote Desktop the following setting should be set:

"TransparentTunnelPortRangeList"="22,1433,3389"

Configuration of tunneling all ports except of 80 and 443 which can be forwarded to iboss using OS-wide configuration looks as follows:
"TransparentTunnelPortRangeList"="1-79,81-442,444-65535"

It is also possible to redirect all ports as follows (which is a default configuration):
"TransparentTunnelPortRangeList"="1-65535"

6 ALTERNATE DEPLOYMENT SCENARIOS

6.1 MANUAL PROXY CONFIGURATION

iboss connector main connectivity option is based on Proxy Autoconfiguration Configuration (PAC) file. In case the application does not support PAC, it is possible to configure proxy manually:

Windows - **geo-cluster125184-swg.ibosscloud.com:8082**

macOS – 127.0.0.1:8009

Linux – 127.0.0.1:8009

Please note that **iboss Connector must still be installed and active** to have a manual configuration working. It is required to register a client in advance.

6.2 GUEST WIFI NETWORKS

Configuration will be provided later. Contact Network Security team for details.

6.3 CONNECTOR-LESS PROXY CONFIGURATIONS

Certain devices may require static proxy configuration without iboss connector. The example of such devices are printers, NAS devices and others. The following proxy definition can be used for internal devices:
proxy-swg-geolb.abb.com:8082
Due to a fact that it is a tedious task to update configuration of such devices, proxy name has been defined under “abb.com” domain is planned to stay even in case of proxy services provider change. Such devices or hosts must use static credentials used for proxy authentication (provided by operations team).

6.4 SITE-WIDE TRANSPARENT PROXY

Locations or segregated networks which contain large number of devices requiring manual proxy configuration or not supporting proxy servers at all may leverage another connectivity option – IPSec VPN tunnel between location and iboss cloud. In this scenario no additional configuration is needed on client devices. All the traffic (ICMP, UDP, TCP) is transparently to iboss for inspection. Contact Network Security team for details.

7 ADDENDUM

7.1 IMPLICATION OF SSL INSPECTION

iboss performs SSL inspection for majority of the traffic (with approved exceptions). The operation involves traffic decryption and re-encryption. This operation may cause issues in certain cases.

7.1.1 Invalid certificate errors

Iboss certificate is signed by ABB internal Root CA. This means any system that has ABB Root CA installed should not get any certificate errors when crossing iboss proxy with HTTPS connection. There may be cases where manual installation is required.

7.1.2 HTTP Public Key Pinningand Certificate Transparency

Some web services use a feature called certificate pinning. This involves additional certificate checks which cause the connection not being established properly. The only solution to that is to request SSL inspection bypass for the URL.

7.1.3 Mutual TLS

Certain web sites require mutual TLS authentication. In such scenario not only server certificate is validated, but also client certificate is, what is not supported by proxy service. The only solution to that is to request SSL inspection bypass for the URL.

7.1.4 Weak ciphers

Some web sites use weak encryption protocols or weak ciphers. ABB protects users by not allowing such protocols to be used (as per GDPR requirements). A web site may be tested using the following service: https://www.ssllabs.com/ssltest/analyze.html. In case this is a business-related site, a whitelisting can be requested.

7.2 OS and Application Specific Notes

All the tools mentioned below can work in Advanced iboss configuration with HTTP Tunnel turned on Windows. In this scenario no proxy configuration is needed.
By default, some Windows applications use OS-wide proxy settings and Linux applications use environmental variables. In many cases however, it is required to configure proxy directly with application or tool configuration file. Examples below cover the most common developer tools.
All the tools mentioned below can work in Advanced iboss configuration with HTTP Tunnel turned on Windows. In this scenario no proxy configuration is needed.
By default, some Windows applications use OS-wide proxy settings and Linux applications use environmental variables. In many cases however, it is required to configure proxy directly with application or tool configuration file. Examples below cover the most common developer tools.

7.2.1 Git

Git requires specific configuration as described here:
https://gist.github.com/evantoli/f8c23a37eb3558ab8765

7.2.2 Pip

Pip can be configured with explicit proxy as described here:
https://pip.readthedocs.io/en/latest/reference/pip/

7.2.3 Anaconda

Anaconda requires specific configuration as described here:
https://conda.io/projects/conda/en/latest/user-guide/configuration/use-condarc.html#configure-conda-for-use-behind-a-proxy-server

7.2.4 Nuget

Nuget proxy configuration can be configured as described here: https://docs.microsoft.com/pl-pl/nuget/reference/nuget-config-file
The following commands add proxy configuration permanently to NuGet config file on **Windows:**
nuget.exe config -set http_proxy=http://geo-cluster125184-swg.ibosscloud.com:8082
Nuget supports also Windows system proxy settings.
The following commands add proxy configuration permanently to NuGet config file on **Linux:**
mono NuGet.exe config -set http_proxy=http://127.0.0.1:8009
Nuget supports also proxy environmental variables on Linux.

7.2.5 Visual Studio Code

VisualStudio Code has the same proxy engine as Chromium project. It should pick up proxy configuration automatically. In case manual proxy configuration is needed, the following command-line arguments can be used (https://code.visualstudio.com/docs/setup/network):
--proxy-server="http://geo-cluster125184-swg.ibosscloud.com:8082"

7.2.6 Visual Studio

VisualStudio should use OS-wide proxy settings on Windows however in case manual configuration is required, the file called devenv.exe.config should be updated (https://docs.microsoft.com/en-US/visualstudio/install/troubleshooting-network-related-errors-in-visual-studio?view=vs-2019):

```
<system.net>
<defaultProxy enabled="true" useDefaultCredentials="true">
<proxy bypassonlocal="True" proxyaddress="http://geo-cluster125184-swg.ibosscloud.com:8082"/>
</defaultProxy>
<system.net>
```

7.2.7 VSTS

VSTS requires the following command to configure proxy (https://docs.microsoft.com/en-us/azure/devops/pipelines/agents/proxy?view=azure-devops&tabs=windows) on Windows:
./config.cmd --proxyurl http://geo-cluster125184-swg.ibosscloud.com:8082
VSTS requires the following command to configure proxy (https://docs.microsoft.com/en-us/azure/devops/pipelines/agents/proxy?view=azure-devops&tabs=unix) on Linux:
./config.sh --proxyurl http://127.0.0.1:8009

7.2.8 Yarn

Yarn can be configured as described here: https://www.jhipster.tech/configuring-a-corporate-proxy/

7.2.9 Npm

Npm can be configured as described here: https://www.freecodecamp.org/forum/t/how-to-run-npm-behind-a-proxy-server-a-step-by-step-guide/19386

7.2.10 Maven

Maven can be configured as described here: https://www.jhipster.tech/configuring-a-corporate-proxy/.

7.2.11 Gradle

Yarn can be configured as described here: https://www.jhipster.tech/configuring-a-corporate-proxy/.

7.2.12 Java

JDK requires additional proxy configuration as described here:
https://docs.oracle.com/javase/6/docs/technotes/guides/net/proxies.html.

JRE work with OS proxy settings or may be configured manually as described here:
https://java.com/en/download/help/proxy_setup.xml.

In case Java app is issuing SSL/TLS connections, importing iboss certificate is required. A short explanation can be found here: https://stackoverflow.com/questions/11617210/how-to-properly-import-a-selfsigned-certificate-into-java-keystore-that-is-avail.

7.2.13 Jenkins

Jenkins requires additional proxy configuration as described here:
https://wiki.jenkins.io/display/JENKINS/JenkinsBehindProxy. Only Server and Port part must be filled in with values provided in Manual proxy configuration

7.2.14 Azure CLI

Azure CLI requires proxy system variable to be set – HTTPS_PROXY=geo-cluster125184-swg.ibosscloud.com:8082. Authentication window required for proper authentication uses standard browser configuration and does not need additional configuration.

7.2.15 .NET applications

.NET applications use different configurations files. Depending on application type, machine.config (system-wide setting), web.config (web application) or app.config (Windows application) can be used. While typically it is not needed to define proxy settings, the manual configuration is possible. The general configuration within .config file is as follows:

```
<defaultProxy
   enabled = "true" [true|false]
   useDefaultCredentials = "false" [true|false]
   />
   <bypasslist>
      <add
         address = "" [String, Required, Collection Key]
      />
   </bypasslist>
   <proxy
      autoDetect = "Unspecified" [False | True | Unspecified]
      scriptLocation = ""
      bypassonlocal = "Unspecified" [False | True | Unspecified]
      proxyaddress = ""
      usesystemdefault = "Unspecified" [False | True | Unspecified]
   />
</defaultProxy>
```

Using this section of the .config file proxy servers can be specified in four different ways.

- autoDetect – will use WPAD protocol to discover proxy configuration
- scriptLocation – allows for entering PAC file URL (in case of iboss this would be http://geo-cluster125184-swg.ibosscloud.com:80/LZahYofGpuVeePp/v2/proxy.pac)
- proxyaddress – manual proxy gateway can be specified here (in case of iboss it would be http://geo-cluster125184-swg.ibosscloud.com:8082)
- usesystemdefault – will use system proxy settings (from WinHTTP configuration, not from IE configuration)

7.2.16 VirtualBox

VirtualBox (just the software, not VM guests inside) needs to have proxy configured manually. In the proxy settings on Windows geo-cluster125184-swg.ibosscloud.com:8082. VirtualBox also does not use operating system certificate store. This means that iboss certificate must be provided to VirtualBox configuration. The certificate can be downloaded from this page [6]. It must be saved as %USERPROFILE%\.Virtualbox\vbox-ssl-cacertificate.crt.

7.2.17 Unity

Unity can use system variables on Windows platform ase as follows:
HTTP_PROXY=geo-cluster125184-swg.ibosscloud.com:8082
HTTPS_PROXY=geo-cluster125184-swg.ibosscloud.com:8082
In addition SSL certificate bundle (can be downloaded from this page [6]) must be stored in folder %ALLUSERSPROFILE%Unity/config. In the same folder configuration file upm-config.json must be created (unless exists already) with the following content:

```
 "caFile": "C:\\ProgramData\\Unity\\config\\ca-bundle-with-iboss.pem"
```

More details on this page: https://docs.unity3d.com/Manual/upm-network.html

7.2.18 Docker Desktop for Windows

Docker Desktop for Windows containers can use advanced configuration of iboss connector and communicate transparently with proxy service. Explicit proxy configuration is also possible (https://docs.docker.com/docker-for-windows/#proxies).

7.2.19 Docker on Linux

Docker on Linux requires special configuration as described here: https://docs.docker.com/network/proxy/.
Example of ~/.docker/config.json file:
"proxies":
"default":
"httpProxy": "http://127.0.0.1:8009",
"httpsProxy": "http://127.0.0.1:8009",
"noProxy": "127.0.0.1"
This configuration will pass proxy information to containers automatically and any application that can use proxy environmental variables (7.1) may leverage them for connectivity.
Docker daemon itself may require additional configuration in /etc/systemd/system/docker.service.d/ folder as described here: https://docs.docker.com/config/daemon/systemd/.
The example configuration in /etc/systemd/system/docker.service.d/http-proxy.conf:
[Service]
Environment="HTTP_PROXY=http://127.0.0.1:8009/" "HTTPS_PROXY=http://127.0.0.1:8009/" "NO_PROXY=localhost,127.0.0.1"

7.2.20 Windows Subsystem for Linux (WSL)

WSL 1 works well with advanced configuration is used based on HTTP Tunneling. It is possible to use explicit proxy configuration either in environmental variables or in tool specific configuration file. For configuration of particular tools (if needed) please look at OS and Application Specific Notes
Majority of tools will require iboss certificate installation within WSL. Installation steps is the same as described here - 7.1.
WSL 2 has not been tested and there is no guarantee it works the same way.

7.2.21 Yum

Yum uses environmental variables but it can also be configured separately as described here: https://www.linuxtechi.com/proxy-settings-yum-command-on-rhel-centos-servers/

7.2.22 Apt

Apt uses environmental variables but it can also be configured separately as described here: https://www.serverlab.ca/tutorials/linux/administration-linux/how-to-set-the-proxy-for-apt-for-ubuntu-18-04/

7.2.23 Curl

Curl can use environmental variables but proxy information can be also specified in a command line.
curl -x "http://127.0.0.1:8009" <URL>

7.2.24 Wget

Wget can use environmental variables but proxy information can be also specified in a command line.
wget -e use_proxy=yes -e http_proxy="http://127.0.0.1:8009" <URL>

7.2.25 Hypervisors (virtual machines)

Local hypervisors such as Hyper-V, VMware, VirtualBox require iboss connector installation within Guest OS. The Host OS iboss installation is not sufficient to provide a proxy connectivity to any of the Guest Operating Systems.

8 LIMITATIONS AND KNOWN ISSUES

– Linux and macOS platforms cannot use HTTP Tunnel mode.
– HTTP Tunnel mode supports only TCP protocol.
– WebSockets protocol works only in case SSL inspection is disabled – this require SSL decryption bypass request for each site/domain.
– Mutual SSL authentication requires SSL inspection to be disabled – this requires SSL decryption bypass request for each site/domain.
– Web sites using certificate pinning protection require SSL decryption bypass request for each site/domain.
– Guest operating systems on client hypervisor require local proxy configuration. They can't leverage host proxy configuration.
– Linux iboss connector does not parse a PAC file properly for proxy bypass list. The fix is pending.

9 APPLICATION TEST RESULTS

WINDOWS

	Default Mode	Advanced Mode	Manual
Git	X	✓	✓
Nuget	✓	✓	✓
VisualStudio Code	✓	✓	✓
VisualStudio	✓	✓	✓
VSTS	X	✓	✓
Jenkins	X	✓	✓
Docker Desktop	X	✓	✓
Docker images	X	✓	✓
Anaconda	X	✓	✓
Yarn	X	✓	✓
Maven	X	✓	✓
Gradle	X	✓	✓
Java	✓	✓	✓
VirtualBox	X	✓	✓
Azure CLI	X	✓	✓
.NET Applications	X	✓	✓
Unity	X	✓	✓
WSL 1	X	✓	✓
WSL 2	X	?	?

LINUX

	Default Mode	Advanced Mode	Manual
Yum	✓	X	✓
Apt	✓	X	✓
Curl	✓	X	✓
Wget	✓	X	✓
Git	X	X	✓
Nuget	✓	X	✓
VSTS	X	X	✓
Npm	X	X	✓
Jenkins	X	X	✓
Anaconda	X	✓	✓
Yarn	X	✓	✓
Maven	X	✓	✓
Gradle	X	✓	✓
Java	✓	✓	✓
Docker	X	X	✓
Docker images	X	X	✓