

APPLICATION EXAMPLE: AE-1078

# AC500 V3 - SMTP

## SENDING EMAIL VIA SMTP PROTOCOL



# Contents

<b>1 Disclaimer .....</b>	<b>3</b>
<b>2 Introduction .....</b>	<b>4</b>
2.1 Compatibility .....	4
2.2 Acronyms .....	4
<b>3 Hardware .....</b>	<b>5</b>
<b>4 Communication principle.....</b>	<b>5</b>
4.1 Communication steps .....	6
<b>5 Mail account .....</b>	<b>7</b>
<b>6 Software .....</b>	<b>9</b>
6.1 File attachments .....	10
<b>7 Troubleshooting .....</b>	<b>11</b>
7.1 Check if the PLC reaches the internet.....	11
7.2 Mail is not received by target mail server.....	11
7.3 StartTLS not supported .....	12

# 1 Disclaimer

A. For customers domiciled outside Germany /

Für Kunden mit Sitz außerhalb Deutschlands

**„Warranty, Liability:**

The user shall be solely responsible for the use of this products described within this file. ABB shall be under no warranty whatsoever. ABB's liability in connection with application of the products or examples provided or the files included within these products, irrespective of the legal ground, shall be excluded. The exclusion of liability shall not apply in the case of intention or gross negligence. The present declaration shall be governed by and construed in accordance with the laws of Switzerland under exclusion of its conflict of laws rules and of the Vienna Convention on the International Sale of Goods (CISG)."

**„Gewährleistung und Haftung:**

Der Nutzer ist allein für die Verwendung des in diesem Dokument beschriebenen Produkte und beschriebenen Anwendungsbeispiele verantwortlich.

ABB unterliegt keiner Gewährleistung. Die Haftung von ABB im Zusammenhang mit diesem Anwendungsbeispiel oder den in dieser Datei enthaltenen Dateien - gleich aus welchem Rechtsgrund - ist ausgeschlossen. Dieser Ausschluss gilt nicht im Falle von Vorsatz oder grober Fahrlässigkeit. Diese Erklärung unterliegt Schweizer Recht unter Ausschluss der Verweisungsnormen und des UN-Kaufrechts (CISG)."

B. Nur für Kunden mit Sitz in Deutschland

**„Gewährleistung und Haftung:**

Die in diesem Dokument beschriebenen Anwendungsbeispiele oder enthaltenen Dateien beschreiben eine mögliche Anwendung der AC500 bzw. zeigen eine mögliche Einsatzart. Sie stellen nur Beispiele für Programmierungen dar, sind aber keine fertigen Lösungen. Eine Gewähr kann nicht übernommen werden.

Der Nutzer ist für die ordnungsgemäße, insbesondere vollständige und fehlerfreie Programmierung der Steuerungen selbst verantwortlich. Im Falle der teilweisen oder ganzen Übernahme der Programmierbeispiele können gegen ABB keine Ansprüche geltend gemacht werden.

Die Haftung von ABB, gleich aus welchem Rechtsgrund, im Zusammenhang mit den Anwendungsbeispielen oder den in dieser Datei enthaltenen Beschreibung wird ausgeschlossen. Der Haftungsausschluss gilt jedoch nicht in Fällen des Vorsatzes, der groben Fahrlässigkeit, bei Ansprüchen nach dem Produkthaftungsgesetz, im Falle der Verletzung des Lebens, des Körpers oder der Gesundheit oder bei schuldhafter Verletzung einer wesentlichen Vertragspflicht. Im Falle der Verletzung einer wesentlichen Vertragspflicht ist die Haftung jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht zugleich ein anderer der in Satz 2 dieses Unterabsatzes erwähnten Fälle gegeben ist. Eine Änderung der Beweislast zum Nachteil des Nutzers ist hiermit nicht verbunden.

Es gilt materielles deutsches Recht unter Ausschluss des UN-Kaufrechts."

## 2 Introduction

### 2.1 Compatibility

The application example explained in this document has been used with the below engineering system versions. They should also work with other versions, nevertheless some small adaptations may be necessary, for future versions.

- AC500 V3 PLC, AC500 V3-eCo PLC (except **PM5012**)
- Automation Builder 2.5.1 or newer



Note:

The mail functionality is only available in library Mail\_Service\_SL version 1.1.3.0 or newer. This Library is supported in Automation Builder 2.5.1 or higher.

Using earlier Automation Builder will not work due to library incompatibility and license terms.

### 2.2 Acronyms

#### **DNS server:**

DNS stands for domain name system and is a nameserver with a static IP address. On the internet, there are static and dynamic IP addresses. The Google DNS for example uses a static IP with '8.8.8.8'. IP addresses from normal consumer devices change within 24h to a new IP address. This is one of the main internet mechanisms, to clear unused or dead IPs.

When typing e.g. [www.abb.com](http://www.abb.com) in your browser, one of the first actions of the computer will be to connect to a name server, resolve the domain by receiving IP address from the server where the website [www.abb.com](http://www.abb.com) is hosted.

If needed, you can use a DNS manually, by opening the CMD command prompt and typing ping [www.abb.com](http://www.abb.com). You will get an answer from the actual IP of [www.abb.com](http://www.abb.com): 'Reply from 172.65.253.13: bytes=32 time=18ms TTL=58'. This IP address is only temporary and only for illustration.

#### **SMTP:**

SMTP stands for Simple Mail Transfer Protocol. The SMTP relay Server is a basic protocol that is used to transfer emails over the Internet. On this server, e.g. smtp.gmail.com, google mail accounts are stored. Clients like browsers or desktop Apps connect to this server and retrieve mails if new ones are available. For retrieving, the POP/IMAP (Post office Protocol /Internet Access Message Protocol) protocol is used. Because it is not used by PLCs, it will not be further discussed here.

#### **SSL:**

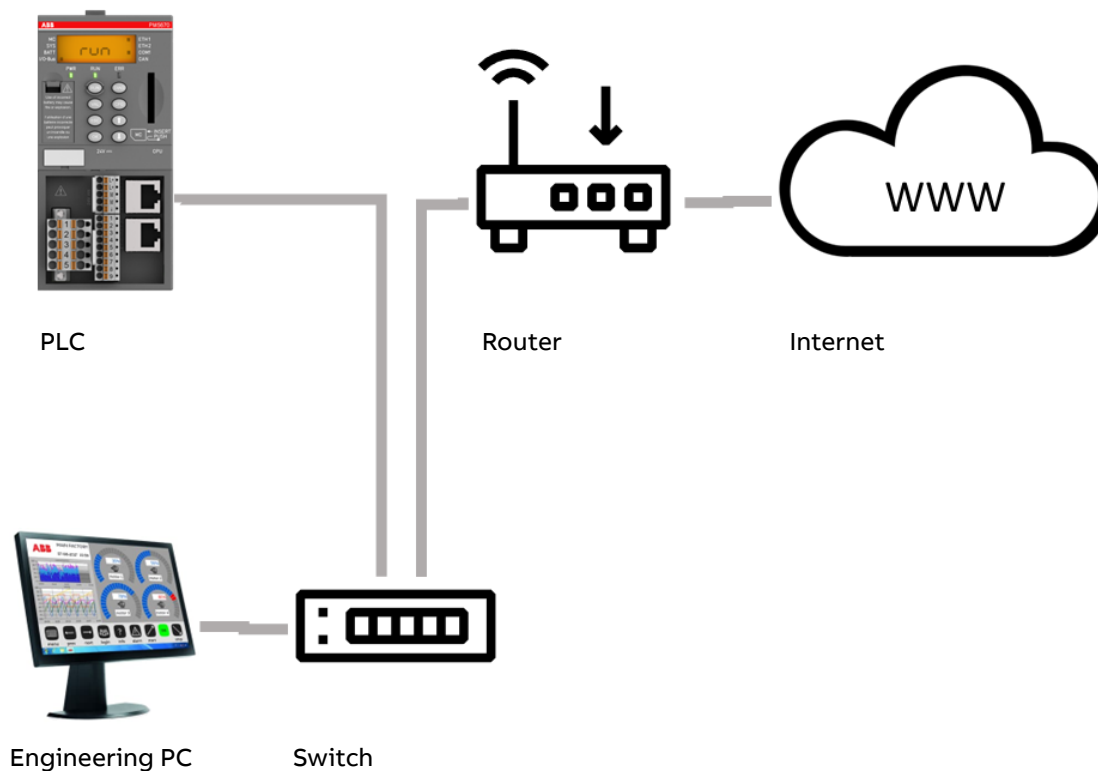
SSL stands for "Secure Sockets Layer" and is the standard technology for securing Internet connections and protecting sensitive data transmitted between two systems.

#### **TLS:**

TLS (Transport Layer Security) is a newer version of SSL that provides a higher security level.

It is often used on the internet for multiple purposes. When your browser URL shows HTTPS then SSL or TLS is used.

### 3 Hardware



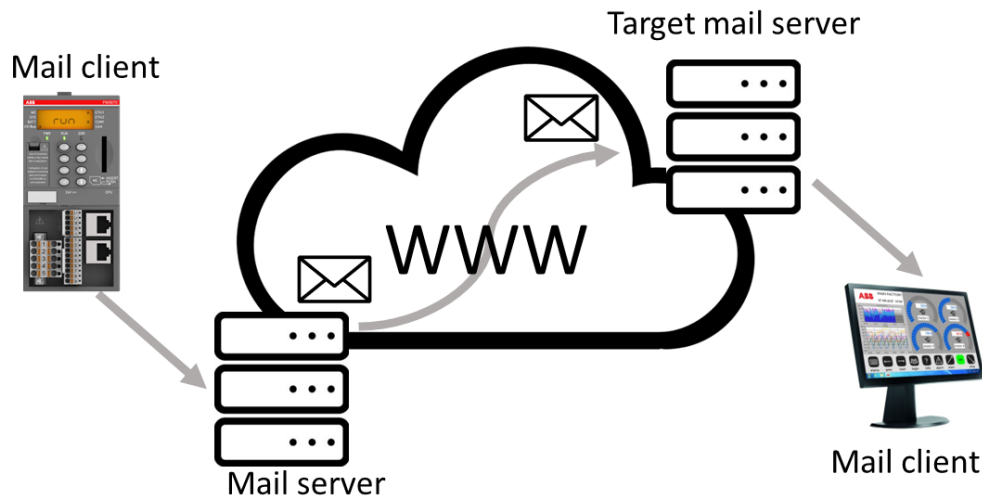
#### Note:

To reach a server on the internet, it's necessary to set the gateway in the PLC. This is normally the IP address of your router.

To test the internet connection, use the PING\_PRG() in the project to ping the Google DNS server with IP: 8.8.8.8

### 4 Communication principle

In this example the AC500 V3 is the SMTP client and needs to be connected to the mail server. This mail server will then send the actual mail. The client itself cannot send a mail on its own. Therefore, it is necessary to create a mail account for this. Also, it is important to set up this mail account correctly, see **Error! Reference source not found..**



## 4.1 Communication steps

### Step 1:

- PLC resolves the IP address of 'smtp.gmail.com' via Google DNS server

### Step 2:

- PLC connects to SMTP Server. In this example smtp.gmail.com
- Uses login parameter to access account.

### Step 3:

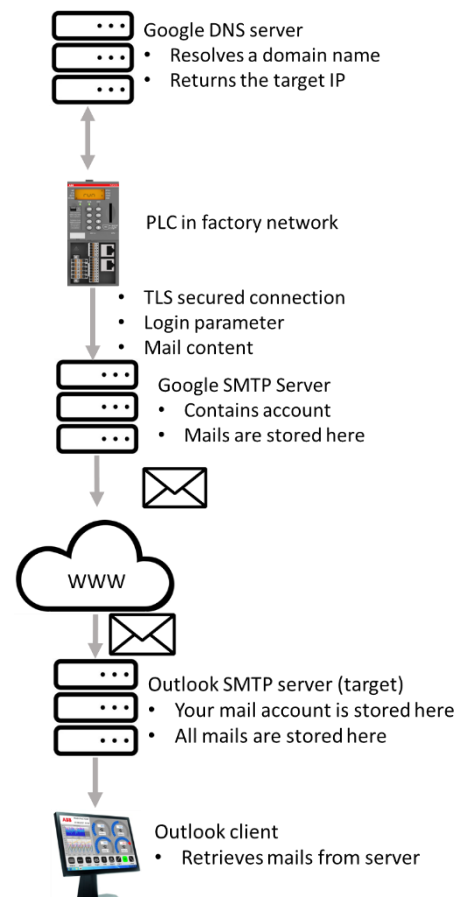
- Google SMTP Server receives all information.
- Stores information in mail account.
- Create and send mail.

### Step 4:

- Mail reaches target SMTP (Outlook) Server.
- Server unpacks the mail and stores the information.

### Step 5:

- Outlook client on your PC refreshes cyclically new mails.
- When a new mail is available at server, client retrieves it and informs user.



## 5 Mail account



### Note:

Nowadays, almost every mail provider uses TLS/StartTLS/SSL encryption to login to the mail account. Additionally, provider like Google use further safety features like checking where the login request is coming from.

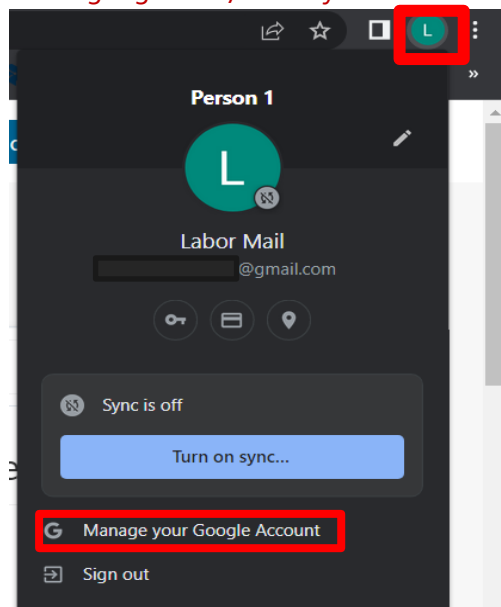
It is important to use the right encryption, credentials, address and port. If the login request fails, the SMTP server gives no response of the error cause.

To be able to use the mail sending functionality, it is mandatory to use a mail account which will send the mails. This should be set up in the first step. It is recommended to use gmail as mail provider as this is most tested. Next, the settings of the account must be modified too.

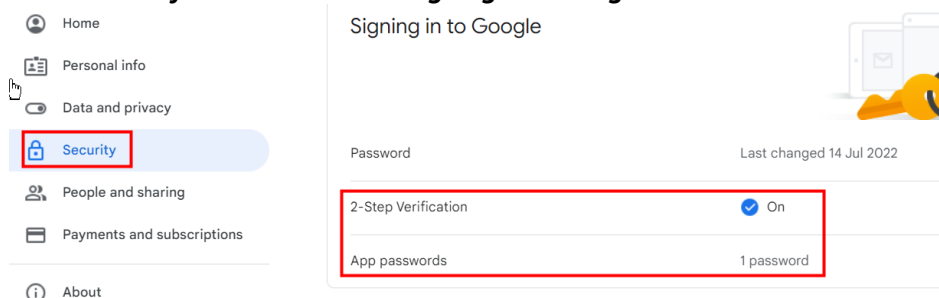
Since end of May 2022, most of the mail provider changed their security policy in terms of third-party devices. To be able to use the SMTP functionality, the account settings must be modified accordingly:

The following instruction guides thru the settings for a Google Mail Account:

- Please navigate to the settings of your Goggle mail account: <https://myaccount.google.com/security>



- Go to **Security** tab and check for **Signing into Google**



- Enable and activate the **2-Step Verification** (follow the wizard to activate the verification with via a mobile number)
- Add a password at **App password** for application purposes. This password will be used in the PLC later on.
  - To access the mail provider via the PLC, the PLC must be added into a list of devices
  - Click on the Dropdown list and select **Other (Custom name)**

App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. [Learn more](#)

You don't have any app passwords.

Select the app and device for which you want to generate the app password.

Mail ▼

- Select device
- iPhone
- iPad
- BlackBerry
- Mac
- Windows Phone
- Windows Computer
- Other (Custom name)**

GENERATE

- By clicking on **GENERATE** a password will be generated which can be used inside the application to connect with the PLC to the mail provider

Generated app password

Your app password for your device

How to use it

Go to the settings for your Google Account in the application or device you are trying to set up. Replace your password with the 16-character password shown above. Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone.

DONE

Email: securesally@gmail.com

Password: .....




**Note:**

The password provided by the App password generator is not the password used to login to the mail account. It is only used to login via an external application to the mail provider.

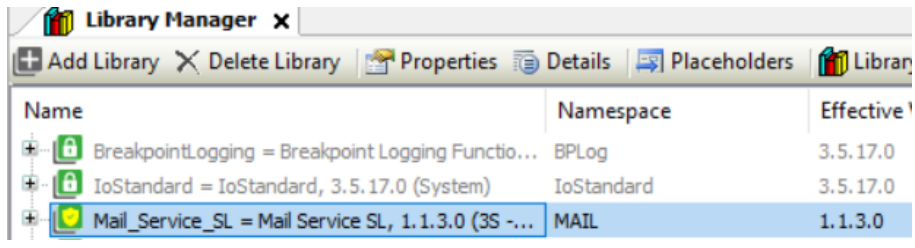


## 6 Software



**Note:**

In this application example, the Library Mail\_Service\_SL (1.1.3.0) is used.



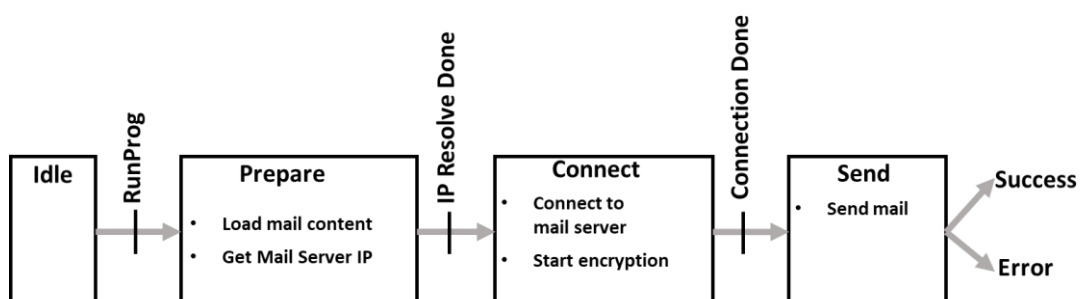
Name	Namespace	Effective Version
BreakpointLogging = Breakpoint Logging Function...	BPLog	3.5.17.0
IoStandard = IoStandard, 3.5.17.0 (System)	IoStandard	3.5.17.0
Mail_Service_SL = Mail Service SL, 1.1.3.0 (3S -...)	MAIL	1.1.3.0

The project contains of the following parts:

- Ping\_PRG()** This program checks for a proper working internet connection. It pings the google DNS server with its static IP address 8.8.8.8
- SmtpClient()** The SmtpClient handles the communication to the mail provider to send an E-Mail with the individual information.

It uses the following routine to send a mail:

- The first step **IDLE** is the default step at program start
- By enabling the RunProg variable the routine starts its process by resolving the IP address of the mail provider using the DNS server
- In the next step **PREPARE**, the communication parameter and mail content are set. The content of the visualization is used. Please take care that the generated password, which is described in *Mail account*, is used instead of the login password of the mail account.
- **CONNECT** will establish a connection to the mail provider
- **SEND** will send the mail with all the previously set parameter





**Note:**

The **udiTimeout** input of the SendMail() function block is mandatory to adjust depending on the size of the mail and internet speed. Using an attachment requires a significantly higher timeout value than without.

```
53      STATE.SEND:
54          fbSendMail(
55              xExecute      := TRUE,
56              udiTimeout    := 120000000,
57              ServerFB      := fbServer,
58              userInfo      := userInfo,
59              mailToSend    := mailToSend,
60              mimeType      :=,
61              sServerMessage => SendMailMessage);
```

## 6.1 File attachments

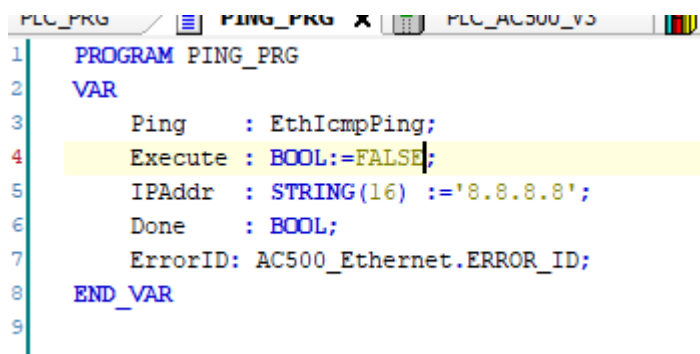
If you want to attach a file, be aware that the sending process takes a certain time. Depending to your file size and your internet upload rate, you have to adjust your timeout in fbSendMail(). The below times refer to the file size with a 2400kbit/s upload speed.

5 Mb	~20 seconds
10 Mb	~30 seconds
20 Mb	~1minute 10seconds
40 Mb	Not possible

## 7 Troubleshooting

### 7.1 Check if the PLC reaches the internet

If you are not sure if the PLC is able to communicate to a SMTP server outside of your factory, you can use the PING\_PRG() functionality included in the example. To execute the ping command, set 'Execute' to TRUE.



```

1  PROGRAM PING_PRG
2  VAR
3      Ping    : EthIcmpPing;
4      Execute : BOOL:=FALSE;
5      IPAddr  : STRING(16) :='8.8.8.8';
6      Done    : BOOL;
7      ErrorID : AC500_Ethernet.ERROR_ID;
8  END_VAR
9

```

### 7.2 Mail is not received by target mail server

This can be caused by different problems. Start by checking the actual step of the step chain:

- State.Idle: Program did not execute the sending progress.
- State.Prepare: If the step chain remains in this step, the DNSResolve function block was not able to resolve the target IP.
  - Please use the PING\_PRG() to ensure the PLC can communicate outside of the factory. Also check your firewall settings, sometimes a ping request is allowed while programs or applications are blocked
  - Make sure to set a valid domain name
  - Check the output ErrorID of DNSResolve for further details
- State.Connect: If the step chain remains in this step, the PLC was not able to access the SMTP server. See variable 'ServerError' for further details. Check your login parameters and port.
- State.Send\_ERROR: The mail could not be sent. See error output from fbSendMail() for further details.
- State.Send\_Success: The mail was successfully sent. If it is not in the Inbox of your mail account check the Spam folder and the settings of the mail account.

## 7.3 StartTLS not supported

```

userInfo
MATI USED INFO

xExecuteFALSE := TRUE,
udiTimeOut 30000000 := 30000000,
ServerFB := fbServer,
userInfo := userInfo,
mailToSend := mailToSend,
mimeType TEXT := types TEXT,
sServerMessage := sServerMessage;

● IF fbSendMail.xDoneFALSE OR fbSendMail.xErrorFALSE THEN
● IF fbSendMail.xDoneFALSE THEN
● sMailSendInfo := 'Your message has been sent';
● MailSuccessfulFALSE := TRUE;
● byStep 100 := 200;
● ELSE
● eSendError STARTTLS_N := fbSendMail.eError NO_ERROR;
● MailSuccess STARTTLS_NOT_SUPPORTED
● byStep 100 'ERROR.STARTTLS_NOT_SUPPORTED' repräsentiert den Rohwert '6001'
● END_IF
● (* This will reset the function block and release the Server lock. *)
● fbSendMail( xExecuteFALSE := FALSE, ServerFB := fbServer);
● END_IF

END_IF

```

The 'eSendError' shows STARTTLS\_NOT\_SUPPORTED. In this case, please check your installed MAIL\_Service\_SL library. It must be at least version 1.1.3.0. Also, subcomponent NetBaseSrv (Net Base Services) must be at least version 3.5.17.0

---

**ABB AG**

Contact:

<https://access.motion.abb.com/contact/contact>

Homepage:

[www.abb.com/plc](http://www.abb.com/plc)

---

We reserve the right to make technical changes or modify the contents of this document without prior notice. With regard to purchase orders, the agreed particulars shall prevail. ABB AG does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

We reserve all rights in this document and in the subject matter and illustrations contained therein. Any reproduction, disclosure to third parties or utilization of its contents – in whole or in parts – is forbidden without prior written consent of ABB AG.  
Copyright© 2024 ABB. All rights reserved.