



# MorphoAccess® SIGMA Series & MorphoWave® Compact

Application note - Distant command using  
TCP client



COPYRIGHT© 2018 IDEMIA

Osny, France

MorphoAccess® SIGMA Series & MorphoWave® Compact – Application note - Distant command using TCP client  
May 2019

2016\_2000022274\_v5

# WARNING

---

COPYRIGHT© 2018 IDEMIA. All rights reserved.

Information in this document is subject to change without notice and do not represent a commitment on the part of IDEMIA. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying or recording, for any purpose without the express written permission of IDEMIA.

This legend is applicable to all pages of this document.

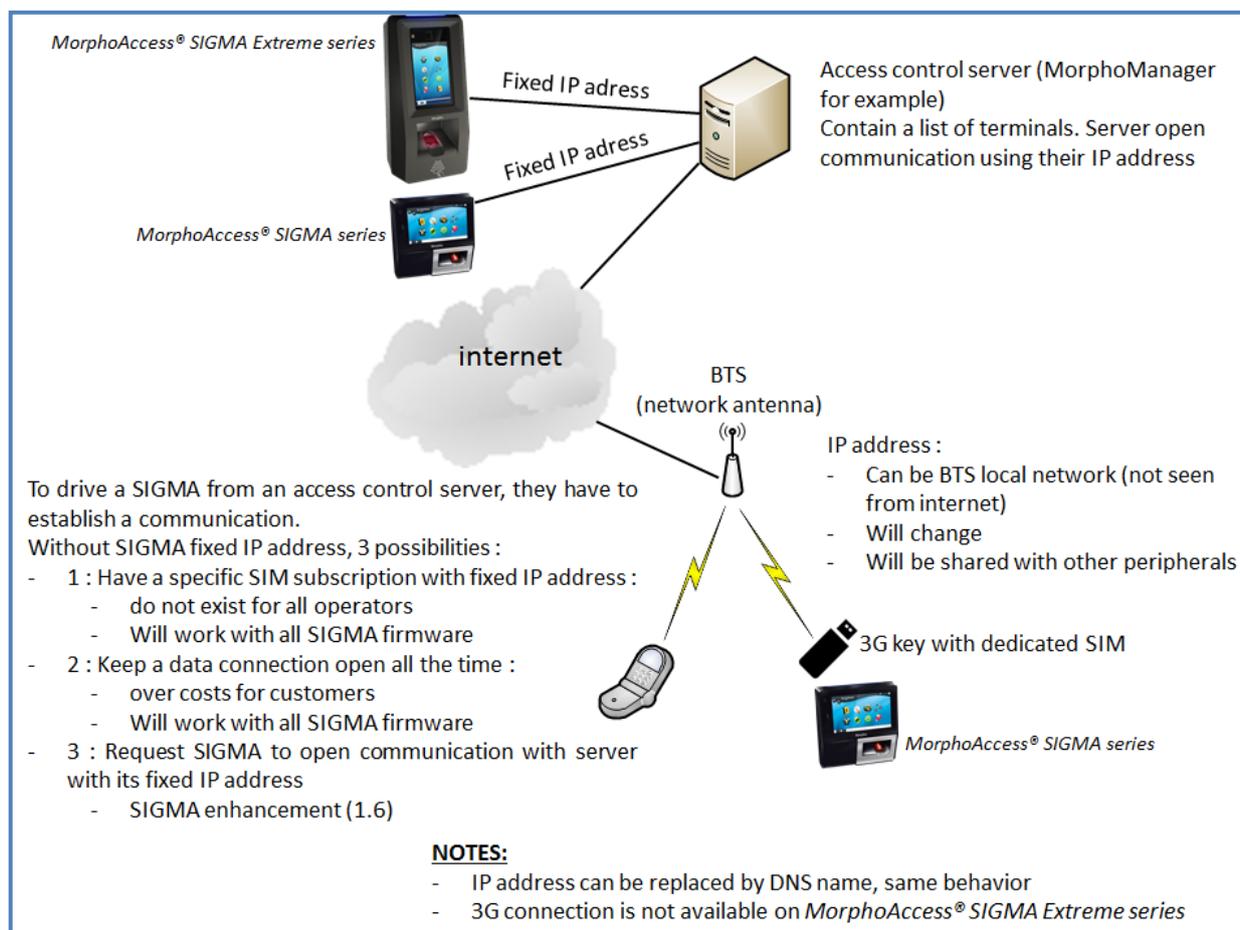
This manual makes reference to names and products that are trademarks of their respective owners.

# Revision History

The table below contains the history of changes made to the present document.

Version	Date	Description
<b>01</b>	October 2016	New document's reference based on 2015_2000011819-V2
<b>02</b>	April 2017	Add <i>MorphoAccess® SIGMA Extreme series</i> support
<b>03</b>	March 2018	Update company name (IDEMIA)
<b>04</b>	July 2018	Add <i>MorphoWave® Compact</i> support
<b>05</b>	May 2019	Add SSL support in TCP client mode. The 2 new keys are : <ul style="list-style-type: none"><li>- TCP_client.SSL_conn_state</li><li>- TCP_client.profile_id</li></ul>

## Distant command using TCP client



## Functionality description

When configured for this function, the reader/the terminal is periodically trying to open a communication channel on a specified IP and port.

When the communication is opened, the reader is waiting, during a configurable time, distant command to execute.

At the end of this duration, without received command, the reader will close the connection.

## Parameters to configure

Reader's administrator should configure the following parameters:

1. **comm\_channels\_state.TCP\_client:**  
Enables / Disables client communication channel which communicates over IP.
2. **TCP\_client.host\_ip:**  
IP address of the machine where server is running.
3. **TCP\_client.host\_port:**  
Mutually decided port number between server and client.
4. **TCP\_client.connection\_period:**  
Period before retrying a connection after previous one ends (failed or succeeded). Period in seconds.
5. **TCP\_client.connection\_timeout:**  
Read/Write/Idle timeout after connection to the server is established. Timeout in seconds.
6. **TCP\_client.connection\_retry:**  
Number of retry on each connection attempt.
7. **TCP\_client.SSL\_conn\_state:**  
Enables / Disables secure communication on TCP client channel.
8. **TCP\_client.profile\_id:**  
Select SSL profile for secure TCP client connection.

### **Warning:**

"**TCP\_client.connection\_timeout**" it should be configured to be higher than the longer command to be potentially executed.

For example if potentially user execute enrolment using distant command,

"**TCP\_client.connection\_timeout**" should be higher than enrolment timeout, else terminal will not be blocked but distant command execution will be failed executed from distant command server.

## Behavior description

Here a configuration example:

```
comm_channels_state.TCP_client: 1
TCP_client.host_ip: 10.126.46.1
TCP_client.host_port: 11020
TCP_client.connection_period: 60
TCP_client.connection_timeout: 10
TCP_client.connection_retry: 1
TCP_client.SSL_conn_state : 0
```

#### **TCP\_client.profile\_id : 0**

1. The reader will try to open connection on port 11020 for the IP 10.126.46.1
2. If it doesn't success to connect, it will directly retry one time
3. If it successfully connects (first or second try), it will execute command received from this connection
4. If it doesn't receive any command during 10s, it will close the connection
5. It will reopen connection 60s after the close
6. It will close again connection after 10s without command
7. Step 4 and 5 are repeated infinitely

#### **Example for Secure TCP client connection:**

The TCP client can be used in secure connection. To enable secure connection for TCP client, the parameter "TCP\_client.SSL\_conn\_state" must be set to 1. By default it is configured to value 0.

The reader will use the SSL/TLS certificate as defined in SSL profile ID ("TCP\_client.profile\_id" parameter) to make a secure connection. By default this parameter is configured to value 0.



Registered Office:

**IDEMIA**

11, boulevard Gallieni

92130 Issy-les-Moulineaux – France

Phone: +33 (0)1 58 11 25 00 – Fax: +33 (0)1 58 11 25 50

<http://www.idemia.com/>