

Secure Messages

What is a "Secure Message"?

A Secure Message is a Short Text Message that can be sent and received securely using PrivateWave . Secure Messages share the same communication infrastructure provided used for secure voice. EVSS do not relay at all on Apple Push Notification or Google Cloud Messaging, in order to protect users' security:

- Server Authentication by SSL certificate.
- HTTPS and SIP over TLS protocols to exchange data in encrypted way.

Secure Messages are text only, thus no attachments are provided. Maximum length of a Secure Message is 160 characters.

How does Secure Message work?

When a user chooses to send a Secure Message, it triggers an HTTPS POST request in his client towards PrivateServer which plays a "store-and-forward" role. If recipient is online, Secure Message is immediately delivered. Otherwise it is stored temporarily in PrivateServer up to 3 days before expiring: as soon as the recipient pops up online, PrivateServer delivers the Secure Message.



Security model provided is end-to-site: message are protected during transport from wiretapping by TLS. Local temporary storage is encrypted with server key.

Delivery is performed by SIP over TLS protocol, so as you can see the whole end-to-end path is encrypted and the message is stored after a local encryption on the server.

Send Secure Messages

It's possible to write Secure Messages to other parties both **by answering** them or **by writing for your own initiative**. In the first case you would probably **use the Contact's History view**, in the latter you would **use the Contact's details view** instead.



Since the "History" lists all the communication attempted or happened on your PrivateWave , it's possible to start a conversation using "History" as well, eg: answering to a missed call by a Secure Message or just using one event conveniently.

Sending Secure Messages from "History" view

As you can see in [History](#) when you look at the screenshots presented as "Example of contact's history event" at the bottom of the Contact's History view, there's a text field you can use to type a new message, like the ones shown below:



figure 1. Typing a new Secure Message in history view (BlackBerry)

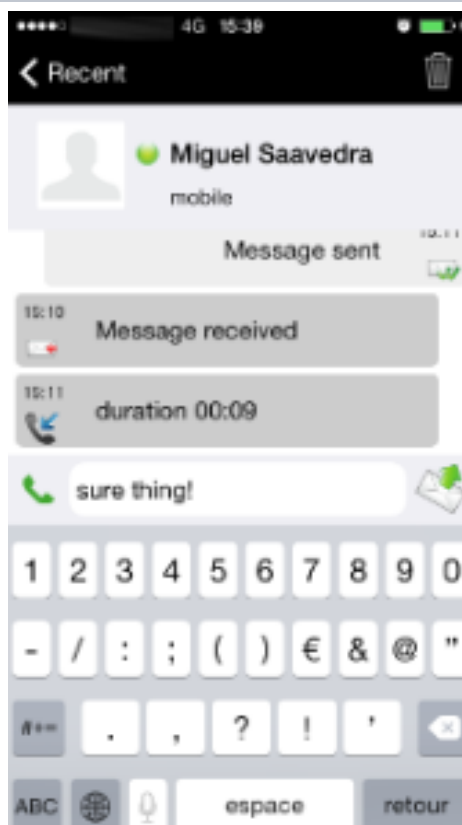


figure 2. Typing a new Secure Message in history view (iPhone)

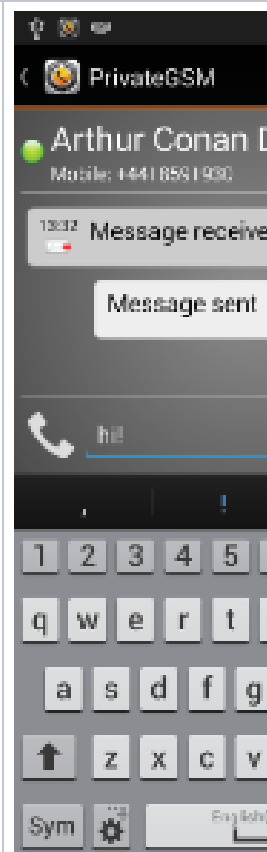


figure 3. Typing a new Secure Message in history view (Android)



Please note that by convention, your outgoing communications are placed on the right side of the screen, while the incoming ones by the chosen contact are placed on the left of the screen.

You can send your new message pressing the proper "Send" button in the User interface if you're on Android or iPhone, else on BlackBerry you just use the "Enter" key in your keyboard.

Sending Secure Messages from "Contact's Detail"

As stated before, Secure Messages are also available from "Contacts" view which is probably the best way to start a conversation via text messages. Obviously first thing it to reach the contact you want to write to. Please refer to [Secure calling](#).

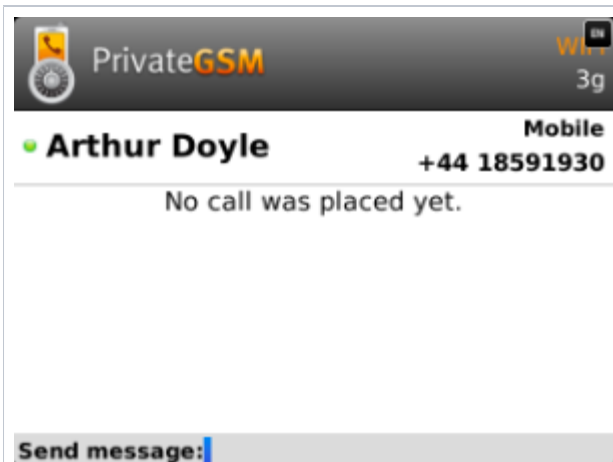


figure 4. New Secure Message from contact's details (BlackBerry)

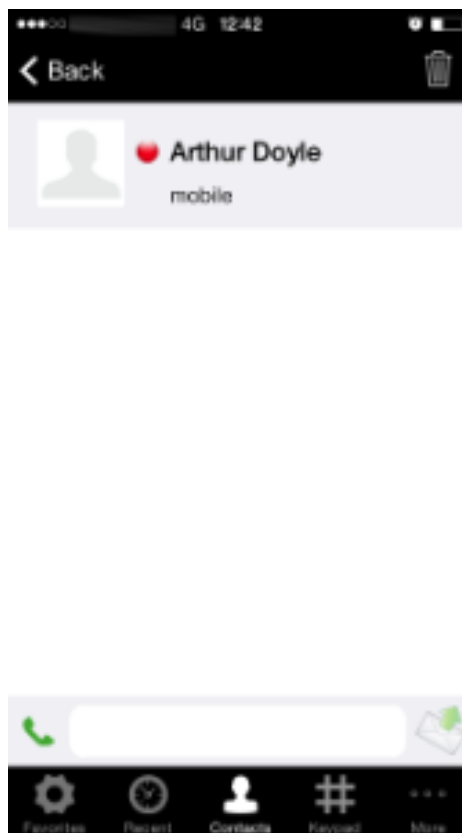


figure 5. New Secure Message from contact's details (iPhone)

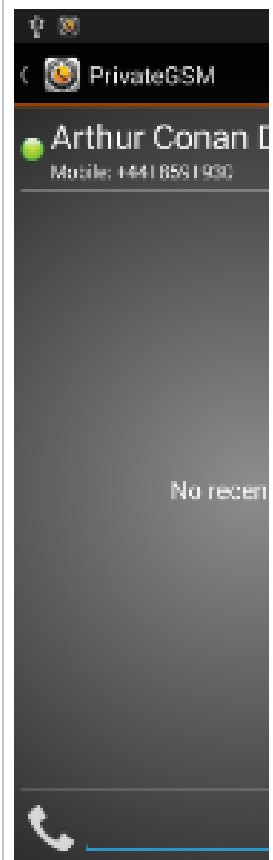



figure 6. New Secure Message from contact's details (Android)

Once you reached the specific number of your peer contact, then you can open his contact's history by pressing the "info" icon in Android and iPhone or just pressing the central button on the keyboard in BlackBerry.

You'll get exactly the same view shown in [History](#) (detailed event view).

 Please note that by convention your outgoing communications are placed on the right side of the screen, while the incoming ones by the chosen contact are placed on the left of the screen.

So you can now type your message, just as in [figure 1. Typing a new Secure Message in history view \(BlackBerry\)](#), [figure 2. Typing a new Secure Message in history view \(iPhone\)](#) and [figure 3. Typing a new Secure Message in history view \(Android\)](#).

Receiving Secure Messages

Obviously a Secure Message can be received as it can be sent. In this case a specific notification is raised by PrivateWave using system's notifications.

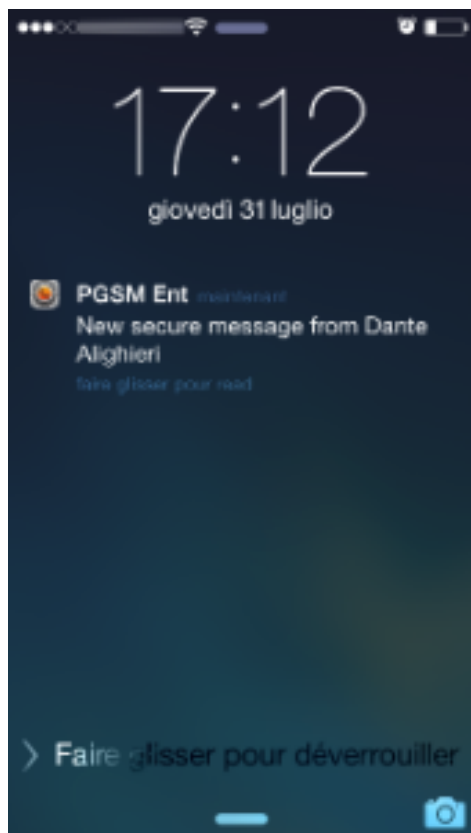


figure 7. new Secure Message notification is shown

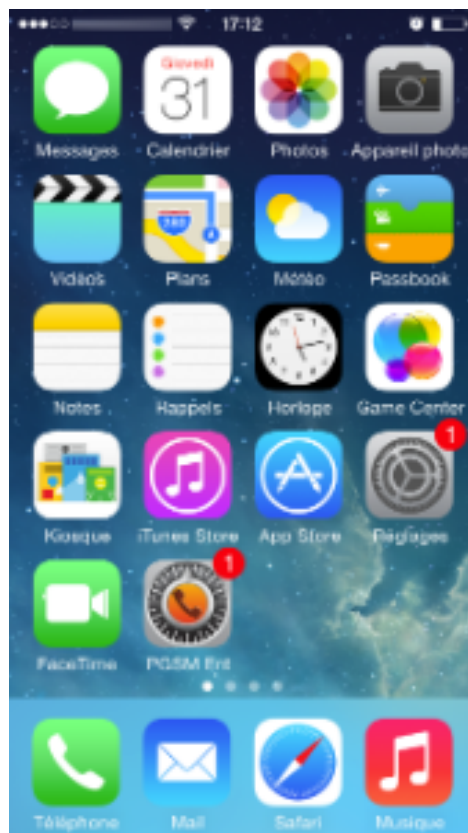


figure 8. event notification on its icon



figure 9. new Secure Message notification is shown

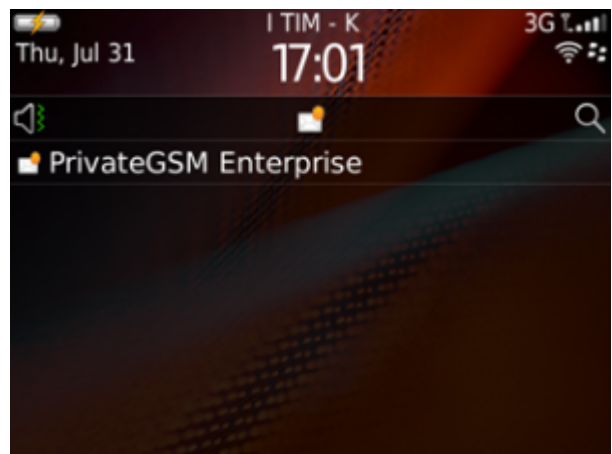


figure 10. drop down event list shows "new message" icon



figure 11. "new message" icon is shown in notification area

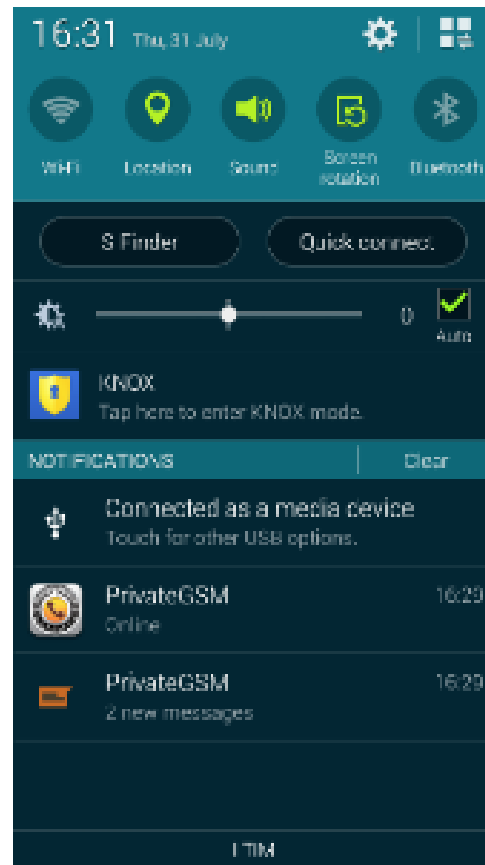


figure 12. drop down event list shows "new message" icon



If just one message has been received, then clicking on the notification would lead you directly to the event detail where you'll be able to read the message.

As shown above each platform shows the event notification differently, according to the underlying operating system behaviour.

Secure Messages compared to regular Text Messages (SMS)

Secure Messages provide a user experience very similar to SMS, adding security and with some minor differences. Some current constraints will be relaxed in next versions.

Secure Messages look just like clear text messages but they are not. Instead they are a sophisticated way for writing each other and their complex architecture implies some **bounds**:

- There's **no chat group** / **no chat room** feature: you can send Secure Messages only to one peer at the time.
- Exception made for BlackBerry, **you can use Emoji** characters to spice up your messages



To use Emoji set it's mandatory to enable this character set on your device. This is an Operating System option thus please refer to your device manual.

- You **cannot attach anything to a Secure Message**: Secure Messages are actually text only.
- Secure Message **length** is **fixed** to a **maximum of 160 characters**.
- Each Secure Message has a **time validity**, after which it **expires**. Expiration means that the Secure Message content is **no longer readable** and it's gone for good. Instead of message content you'd rather read "Message received" or "Message sent".



figure 13. BlackBerry expired Secure Message

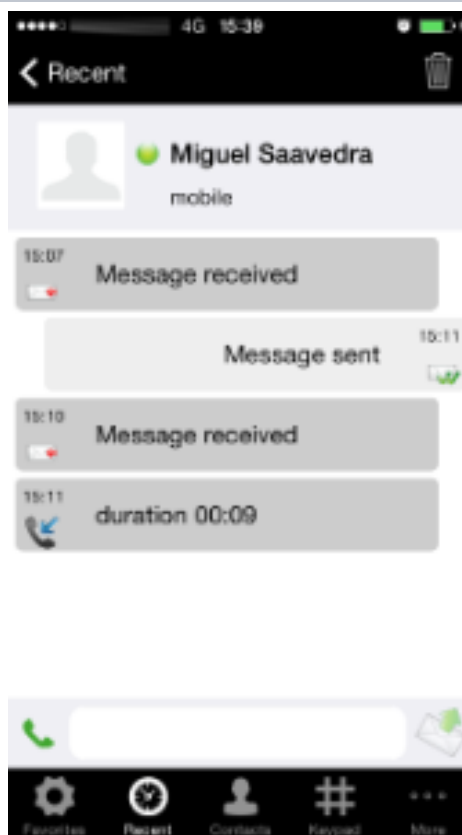


figure 14. iPhone expired Secure Message



figure 15. Android Mess:

Each time you send a Secure Message you can see a status icon appearing on its side. Here follows an explanation of each icon along with its status name and meaning:

| Icon | Description | Meaning | Timeouts and general behaviour |
|------|-----------------------|---|--|
| | Sending Message | PrivateWave is actually trying to send the Secure Message to its recipient via PrivateServer | <p>If there is some network issue, then PrivateWave starts the retry procedure: it tries to send 5 times the message to PrivateServer . If it fails at the fifth try (5 * TLSTimeout = 60 secs), then the message remains in the client queue waiting for one of the following triggers:</p> <ul style="list-style-type: none"> ◦ The user can connect to Sip Server ◦ The client sends a SIP KeepAlive to the server ◦ The user sends another message <p>If more than 12 hours passed and the message is still in client queue waiting to be sent, then the message expires: a "Send error" notification is fired by PrivateWave and the message is removed from local queue</p> |
| | Message Enqueued | Secure Message has been enqueued in PrivateServer encrypted storage and will be delivered to recipient as soon as possible. | If delivery is not possible because the peer is offline, PrivateServer retries as soon as recipient registers for up to an expiration period (72h), before removing them definitely from the database. |
| | Message Delivered | Secure Message has been delivered to recipient's device, who could have read it or not. | None |
| | Error sending Message | This status is a fall back for any other one and it's triggered both by timeouts and error responses. | None |

[History](#)

[Other functions and settings](#)

