Wie funktioniert GoCrypt?

GoCrypt verwendet einen asymmetrischen kryptographischen Algorithmus - RSA (https://en.wikipedia.org/wiki/RSA_(cryptosystem)), um die von Ihnen gesendete Datei zu verschlüsseln/entschlüsseln. Die Besonderheit und der Vorteil des RSA-Algorithmus ist, dass es sich um einen so genannten asymmetrischen Algorithmus handelt, d.h. es gibt ein Schlüsselpaar, das in einem Prozess verwendet wird - den öffentlichen Schlüssel (für die Verschlüsselung der Datei) und den privaten Schlüssel (für die Entschlüsselung). Die Namen der Schlüssel bestimmen ihre Verwendung, d.h. der öffentliche Schlüssel kann an jeden weitergegeben werden, es gibt keine Schwachstelle, der private Schlüssel muss jedoch geheim gehalten werden. Bitte beachten Sie, dass es bei Verlust des privaten Schlüssels unmöglich ist, ihn irgendwie zurückzubekommen, also bewahren Sie ihn sorgfältig auf, um ihn nicht zu verlieren.

Das RSA-Schlüsselpaar wird bei der Registrierung im GoCrypt-System generiert, der öffentliche Schlüssel wird an den Server gesendet, wo er dauerhaft gespeichert wird, der private Schlüssel wird geheim auf Ihrem Computer (in einer Version ohne Token) oder auf einem Token aufbewahrt, beide passwortgeschützt.

Beim Versenden der Datei benötigt GoCrypt den öffentlichen Schlüssel des Empfängers (der auf dem Server gespeichert ist), um den Dateiinhalt zu verschlüsseln. Nach der Verschlüsselung wird die Datei auf den Server geladen, wo sie vorübergehend gespeichert wird. Die Datei verlässt den Computer des Absenders nie unverschlüsselt, was garantiert, dass keine Daten irgendwie gestohlen oder weitergegeben werden können. Wenn die Datei auf den Server hochgeladen ist, erhält der Empfänger eine Benachrichtigung über die Datei, die auf das Herunterladen wartet. Die verschlüsselte Datei wird vom Server heruntergeladen und mit dem privaten Schlüssel des Empfängers entschlüsselt. Der private Schlüssel des Empfängers verlässt niemals den Computer (bei der Version ohne Token), bei der Version mit Token ist es sogar noch sicherer, da die Entschlüsselung innerhalb des Tokens stattfindet und GoCrypt in diesem Fall keinen Zugang zum privaten Schlüssel hat.

Prozess der Registrierung

Während der Registrierung generiert GoCrypt ein einzigartiges Paar von privaten und öffentlichen Schlüsseln für den Kunden. Der öffentliche Schlüssel wird an den Server gesendet und dort dauerhaft gespeichert, der private Schlüssel wird sicher auf Ihrem Computer (bei der Version ohne Token) oder in einem Token gespeichert. Obligatorische Aktion für die Registrierung - E-Mail-Bestätigung. Sobald das Schlüsselpaar generiert ist, sendet der Server eine automatische E-Mail-Nachricht an die bei der Registrierung angegebene E-Mail-Adresse mit einem Link zur Bestätigung der E-Mail-Adresse. Sie müssen den Eingangsordner auf E-Mails von noreply@gocrypt.de überprüfen (möglicherweise müssen Sie den Spam-Ordner überprüfen) und den Link zur E-Mail-Bestätigung (Kontoaktivierung) in einem Browser öffnen (Link in einen Browser kopieren/einfügen oder anklicken, um ihn automatisch im Standardbrowser zu öffnen). Wenn der Bestätigungsprozess gut durchgeführt wurde, sollte die erfolgreiche Webseite in einem Browser angezeigt werden.

Sicherheit QA

F: Ist es möglich, dass jemand meinen privaten Schlüssel stiehlt?
A: **Für die Version ohne Token:** Der private Schlüssel ist auf Ihrem Computer gespeichert, aber er ist mit einem Passwort verschlüsselt. Ohne das Passwort ist der private Schlüssel nutzlos.
Bei der Version mit FidoToken: Der private Schlüssel befindet sich im Fido Token und verlässt ihn nie. Selbst wenn jemand physischen Zugang zu Ihrem Fido Token erhält, kann er nicht verwendet werden, solange er das Fido Token-Passwort nicht kennen.

F: Ist es möglich, dass die von mir gesendete/heruntergeladene Datei von einem Sniffer (Tool zur Verkehrsanalyse) gestohlen wird?
A: Nein, das ist unmöglich. Die von Ihnen gesendete/heruntergeladene Datei verlässt/erreicht Ihren Computer nie entschlüsselt. Die Verschlüsselung erfolgt vor dem Senden und die Entschlüsselung nach dem Herunterladen. Es macht also keinen Sinn, den Datenverkehr zu analysieren, die Dateien kommen nur verschlüsselt durch das Netzwerk.

F: Ich werde Windows neu installieren, wird das die GoCrypt-Installation beeinflussen?
A: **Für die Version ohne Token:** Der private Schlüssel wird in der Windows-Registrierung gespeichert und muss während eines Windows-Updates (Installation eines Updates oder eines größeren Updates wie die Migration von Windows 10 auf Windows 11) beibehalten werden. Wenn es sich um eine saubere Installation handelt, dann müssen Sie sich erneut in GoCrypt registrieren.
Für die Version mit Fido Token: Änderungen von Windows haben keinen Einfluss auf GoCrypt, da der Fido Token zum Speichern des privaten Schlüssels verwendet wird.

Technische Informationen

Server-Kommunikation

GoCrypt kommuniziert mit dem Server über das GRPC-Protokoll/Framework. Die Serverseite bietet eine Reihe von Funktionen (Server-API) für die Kommunikation mit dem Server. Für einige der Server-API-Funktionen ist eine Authentifizierung erforderlich, die nach Ablauf des JWT-Tokens (siehe Authentifizierungsablauf unten) beim nächsten API-Aufruf durchgeführt wird.

Anmeldung

Die Registrierung erfolgt als eine Folge der folgenden Ereignisse:

- GC sammelt Informationen aus einem Registrierungsformular und sendet die Registrierungsanfrage an den Server (über die Server-API);
- Server, der die Registrierungsanfrage annimmt, erzeugt eine Antwort, die eine Nonce enthält. Nonce - zufällig generiertes 32-Byte-Array;
- GC generiert ein RSA 2048-Schlüsselpaar (je nach Version verwendet GC entweder .NET Framework-Funktionen, um ein Schlüsselpaar zu generieren, oder ruft das Token SDK auf, um die Generierung von Schlüsseln auf einem Token zu erzwingen);
- GC signiert die von einem Server erhaltene Registrierungs-Nonce mit dem generierten privaten Schlüssel und sendet die signierte Nonce + den öffentlichen Schlüssel an einen Server, um die Registrierung abzuschließen;

- Der Server erhält eine vollständige Registrierungsanfrage, die eine signierte Nonce und den öffentlichen Schlüssel des Nutzers enthält. Die Nonce-Signatur wird zunächst anhand des öffentlichen Schlüssels überprüft, ob die Signatur gültig ist (diese Aktion beweist, dass die Nonce mit dem privaten Schlüssel signiert wurde, der mit dem erhaltenen öffentlichen Schlüssel verknüpft ist), und dann wird die Nonce auf Korrektheit und zeitliche Gültigkeit überprüft (die Nonce ist nur für einen kurzen Zeitraum gültig, um zu verhindern, dass sie irgendwie geknackt oder manipuliert wird). Wenn nonce die Validierungsregeln bestanden hat, sendet der Server eine Bestätigungsnachricht an die bei der Registrierung angegebene E-Mail-Adresse.
- Um mit der Nutzung von GoCrypt zu beginnen, muss der Benutzer eine E-Mail mit einem Aktivierungslink (Bestätigung) erhalten. Dieser Link sollte im Browser geöffnet werden, um zu bestätigen, dass der Nutzer diese E-Mail-Adresse besitzt. Um Manipulationen mit dem Aktivierungslink zu vermeiden, hat dieser auch eine Verfallszeit von 3 Stunden ab dem Zeitpunkt der Registrierung;

Authentifizierung

Der Authentifizierungsprozess ist ähnlich wie die Registrierung und basiert auf einer Nonce-Signatur. Der Ablauf ist der folgende:

- GC sendet Authentifizierungsanfragen, die die E-Mail-Adresse des registrierten Kontos enthalten;
- Der Server überprüft die E-Mail-Adresse und den zugewiesenen Benutzer, generiert ein Nonce - ein Array aus 32 Zufallsbytes - und sendet es an GC zurück.
- GC signiert die Nonce mit dem privaten Schlüssel und sendet sie als vollständige Authentifizierungsanfrage an den Server zurück;
- Der Server validiert die signierte Nonce anhand des auf dem Server gespeicherten öffentlichen Schlüssels für diesen Benutzer. Ist die Signatur korrekt, validiert er auch die Nonce, sofern sie korrekt und noch nicht abgelaufen ist. Wenn die Nonce-Prüfung erfolgreich war, erstellt der Server ein JWT-Token mit dem Algorithmus HMAC-SHA256. Das JWT-Token ist 5 Minuten lang gültig, nach Ablauf des Tokens wird das Authentifizierungsverfahren wiederholt;
- GC erhält das JWT-Token, um es für die Serverkommunikation während der Lebensdauer des Tokens zu verwenden;

Verschlüsselung

Dateioperationen mit der Server-Api werden über einen authentifizierten Kanal durchgeführt. Der unten beschriebene Zyklus wiederholt sich für jede gesendete Datei:

- GC ruft den Server auf, um eine Anfrage zum Hochladen von Dateien zu erstellen;
- GC erzeugt einen AES-Schlüssel von 256 Bit und einen Init-Vektorschlüssel;
- GC berechnet den SHA256-Hash der Originaldatei;
- Die zu sendende Datei wird mit einem AES-Schlüssel verschlüsselt;
- Der AES-Schlüssel wird mit dem öffentlichen RSA-Schlüssel des Empfängers verschlüsselt.
- Der Hash der Originaldatei, die verschlüsselte Datei, der verschlüsselte AES-Schlüssel und der AES-Init-Vektor werden an den Server gesendet.

Entschlüsselung

Dateioperationen mit der Server-Api werden über einen authentifizierten Kanal durchgeführt. Der unten beschriebene Zyklus wiederholt sich für jede gesendete Datei:

- GC ruft den Server an, um eine Anfrage zum Herunterladen von Dateien zu erstellen;
- Der Server antwortet und sendet dem GC die verschlüsselte Datei, den verschlüsselten AES-Schlüssel und den AES-Init-Vektor zurück;
- GC entschlüsselt mit dem privaten Schlüssel des Empfängers den AES-Schlüssel. Mit dem AES-Schlüssel und dem Init-Vektor entschlüsselt GC die Datei;
- GC berechnet den Datei-Hash der entschlüsselten Datei und ruft den Server beim Herunterladen der vollständigen Anfrage an;
- Der Server prüft, ob der Hash der Originaldatei mit dem Hash der entschlüsselten Datei des Empfängers übereinstimmt. Wenn die Hashes übereinstimmen, markiert der Server die Datei als heruntergeladen und erlaubt GC, die Datei am gewünschten Ort zu speichern.

Stand 11/2023