Elliptic Curves & Academic Signature

von Johannes Steinkamp

Inhaltsverzeichnis

Contents

1. Elliptic Curves

- Algebraische Kurven:

Auf einer algebraischen Kurve ist geometrisch eine Addition definiert. Diese Kurve wird als eine glatte, algebraische Kurve in der projektiven Ebene definiert. Dargestellt wird diese meist als Kurve in der affinen Ebene. Charakteristisch für die affine Ebene ist, dass je zwei Punkte eine Verbindungsgrade haben und dass eindeutig parallele Geraden vorhanden sind.

2. Elliptic Curve als Funktion

- Funktion:

Elliptische Kurven können als die Menge aller Punkte, die ein Element aller reellen Zahlen sind, angesehen werden. Elliptische Kurven erfüllen folgende Gleichung:

$$y^2 = x^3 + ax + b$$

Zusätzlich besitzen elliptische Kurven einen Punkt im Unendlichen. Die Bedingung hierbei ist:

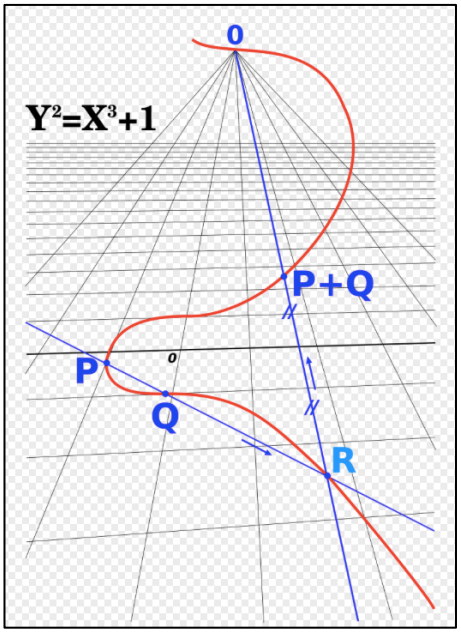
$$4a^3 + 27b^2 \neq 0$$

- Rechenregeln für die Addition zweier Punkte:

1. $P + Q = Q + P$
2. $P + (-P) = \infty$
3. $P + \infty = P$
4. $(P + Q) + R = P + (Q + R)$

- Addition zweier Punkte:

Der Punkt O beschreibt hierbei im Unendlichen das neutrale Element. Die Addition erfolgt zunächst durch das Ziehen einer Gerade durch die zwei Punkte P und Q. Der hierbei entstandene Schnittpunkt der Geraden mit der elliptischen Kurve wird an der x-Achse gespiegelt. Dadurch ergibt sich der neue, gespiegelte Punkt P+Q. Dies ist der Punkt, der der Addition der einzelnen Punkte P und Q entspricht.



3. BEGRIFFE DER KRYPTOGRAFIE

- Kryptografie:

Der Begriff „Kryptografie“ stammt aus dem griechischen Sprachgebrauch. „Krypto“ übersetzt bedeutet „verborgen“ und „graphein“ schreiben. Zusammengesetzt bedeutet es also verborgen schreiben bzw. verborgen kommunizieren. Dies passiert ohne, dass es ein Dritter mitbekommt.
Die Kryptographie ist somit die Wissenschaft der verborgenen Kommunikation und der Ver- und Entschlüsselung von Informationen.

- Domain:

Eine Domäne ist eine lokale Plattform, die als eine logische Einheit betrachtet wird. Der Domain-Name gibt beispielsweise den eindeutigen Namen einer Internetwebsite wieder. Der Domain-Name unterteilt sich in diverse Ebenen. Die ersten beiden Ebenen einer Domäne werden Top-Level-Domain (TLD) und Second-Level-Domain genannt. Die Third-Level-Domain ist dann die erste Subdomain.
Beispiel:
https://de.wikipedia.org
TLD: org
SLD: wikipedia

- Hashfunktion:

Der Hintergrund der Hashfunktion ist, dass sie eine große Eingabemenge (Schlüssel) auf eine kleinere Zielmenge (Hashwerte) abbildet. Es lässt sich so vorstellen, dass man eine unbestimmte Menge hat und diese möglichst gerecht auf beispielsweise Schüsseln verteilt wird.

- Chimera:

Chimera ist ein Algorithmus, der agiert, wie es der Name bereits verspricht. Er beinhaltet zwei Algorithmen, Threefish und Flightx. Die beiden Algorithmen zusammen wirken entgegengesetzt. Das bedeutet, wenn Threefish geknackt wird, muss immer noch Flightx gleichzeitig geknackt werden. Es reicht hierbei nicht, dass einer von beiden gehackt wird. Diese Verbindung macht Chimera aus.

- Entropie:

Entropie beschreibt die Informationsdichte einer Nachricht

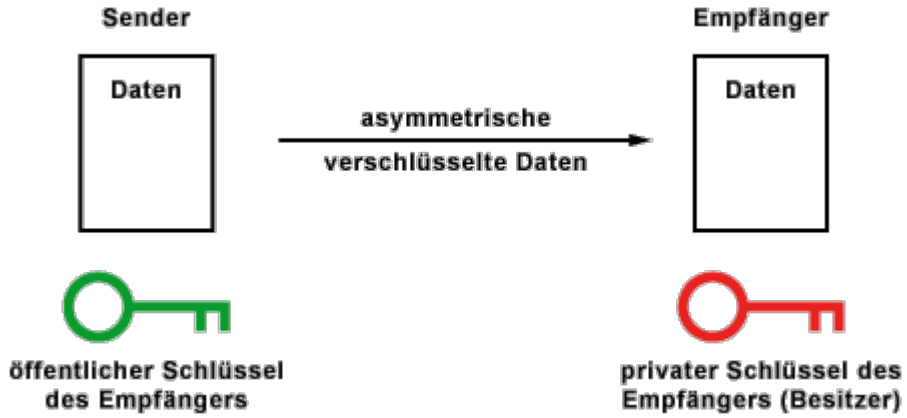
4. ACADEMIC SIGNATURE

WAS IST ACADEMIC SIGNATURE?

- Academic Signature ist ein Programm der Asymmetrischen Kryptografie, um Dateien/ Nachrichten zu ver- und entschlüsseln.

WAS IST DIE ASYMMETRISCHE KRYPTOGRAFIE?

- Die asymmetrische Kryptografie wird auch Public-Key-Verfahren genannt und ist eine Verschlüsselung, bei der man mit einem Schlüsselpaar, bestehend aus öffentlichem und privatem Schlüssel (Schlüsselpaar), Dateien/ Nachrichten verschlüsselt.



ZUSATZINFORMATION - SYMMETRISCHE KRYPTOGRAFIE

- Die symmetrische Kryptografie hat im Gegensatz zur asymmetrischen nur einen geheimen Schlüssel, den beide sowohl zur Verschlüsselung als auch zur Entschlüsselung nutzen. Problematisch ist hierbei immer die Übergabe des geheimen Schlüssels.

5. Funktionen von Academic Signature

- Zum einen kann man mittels Academic Signature mit dem elliptischen Kurvenverfahren Dateien codieren/ chiffrieren und ebenso wieder decodieren/ dechiffrieren.
- Zum anderen können elektronische Signaturen erstellt und verifiziert werden

Vorgehensweise bei Versenden einer verschlüsselten Datei/ Nachricht:

Beispiel 1: Bob möchte Alice eine verschlüsselte Datei/ Nachricht senden:

- Alice erstellt einen privaten Schlüssel, der vom User an einem sicheren Ort und geheim gehalten werden sollte und zusätzlich einen öffentlichen Schlüssel. Diese beiden Schlüssel werden zusammen auch Schlüsselpaar genannt.
- Der öffentliche Schlüssel kann von Alice an einem öffentlich zugänglichen, medialen Ort abgelegt werden (z.B. auf der Mitarbeiterseite der FH Wedel).
- Bob erfährt nun über die Mitarbeiterseite der FH Wedel den öffentlichen Schlüssel von Alice und chiffriert mit dem public key die Datei/ Nachricht
- Die Datei/ Nachricht ist nun verschlüsselt und kann nur (Bedingung: der geheime Schlüssel ist an keinen Dritten gelangt, da Alice sehr vorsichtig mit seinem geheimen Schlüssel umgegangen ist) von Alice mit seinem privaten Schlüssel dechiffriert werden.

Beispiel 2.1: Alice möchte einer Datei/ Nachricht eine elektronische Signatur anhängen:

- Eine elektronische Signatur kann von Alice nur mit seinem geheimen Schlüssel erzeugt werden
- Alice sendet Bob eine Datei/ Nachricht, der eine elektronische Signatur angehängt ist
- Bob kann nun mit Hilfe des öffentlichen Schlüssels die Signatur von Alice überprüfen

Beispiel 2.2: Bob leitet Datei/ Nachricht weiter:

Desweiteren könnte Bob die Datei/ Nachricht weiterleiten und die Empfänger könnten dann ebenfalls mit dem öffentlichen Schlüssel von Alice überprüfen, ob die Datei von Alice signiert wurde und somit kann auch geprüft werden, ob Bob etwas an der Datei verändert hat.

6. KRYPTOPARTY – INSTALLATION

1. Für Windows XP, 7, 8 User:
   Unter folgendem Hyperlink http://www.fh-wedel.de/~an/crypto/Academic_signature_eng.html kann man entweder
   A) die intelligente Version, die Updates inkludiert, oder
   B) die einfach gehaltene Version (für die, die das Passwort oft vergessen)
   von Academic Signature herunterladen.

Die Windows-Vollversion(für XP, 7,8) wird in zwei Installationsvarianten angeboten:
Leider scheint nicht jedes Windows10 trotz anderslautender Behauptungen von Microsoft in der Lage zu sein, wxWidgets basierte Windows7/8 Programme auszuführen. (Meistens geht's trotzdem problemlos.)
Neu! Jetzt gibt es eine dedizierte Windoes10 Variante -> siehe unten.

A) Die "*intelligente*" Version.
(Dies ist die Version für engagierte Nutzer, die jetzt oder später intelligente Updates ausführen möchten)
Der Installer der setup_update_Version schreibt neben der Programmdatei nur Files auf die Festplatte, die keine nutzerspezifischen Daten überschreiben. Vorhandene Passwort Informationen und Schlüsselfiles werden respektiert und belassen. Nach dem Setup wird bei Erstinstallation der Schlüpf- und Einprägevorgang gestartet, andernfalls wird bei einem normalen Login das vertraute Passwort abgefragt und alle bisherigen Schlüsselinformationen stehen weiter zur Verfügung.
Der Deinstaller ist folgerichtig weniger ruppig und belässt nutzerspezifische Information im System, so dass diese bei einer späteren Reinstallation wieder genutzt werden kann.
aca_sig_setup_update , GnuPG Signatur , ECDSA-Signatur

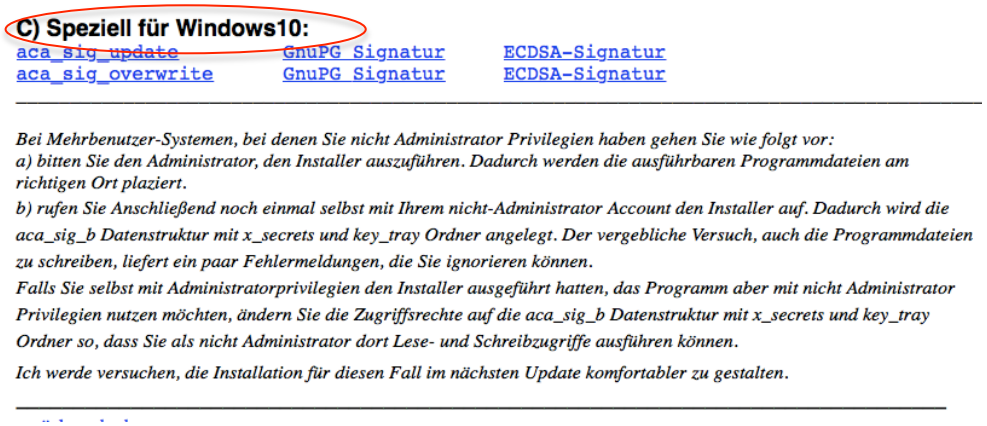
Auf vielfachen Wunsch jetzt auch in deutschsprachiger Version:
aca_sig_setup_update_de , GnuPG Signatur , ECDSA-Signatur

(Wenn Sie nach Installation unerwünschterweise nach einem vergessenen Passwort gefragt werden, können Sie auch manuell das Verzeichnis "x_secrets" leeren oder entfernen und dann erneut den Setup ausführen. Als Hinweis gibt das Programm nach dem dritten fehlgeschlagenen Login-Versuch den Ort aus, an dem Windows dieses Verzeichnis abgelegt hat.)

B) Die "*Weg da hier komm ich*" Version.
(Dies ist die Version für die Schafsnasen, die Ihr Passwort vergessen werden oder schon vergessen haben.....
alle anderen sollten die update Version wählen)
Der Installer der setup_overwrite_Version drückt alle benötigten Files an die richtige Stelle. Wenn da noch alte Files einer vorhergehenden Installation standen - Pech gehabt. Alle alten Informationen werden überschrieben(Salts, Schlüssel, Einstellungen, Passwort Hash...) und sind unwiederbringlich weg. Beim ersten Aufruf wird die "Schlüpf- und Einprägeroutine" aufgerufen.
Der Deinstaller ist folgerichtig genauso brutal und reisst neben der Programmdatei auch alle nutzerspezifischen Informationen mit heraus.
aca_sig_setup_overwrite , GnuPG Signatur , ECDSA-Signatur

Auf vielfachen Wunsch jetzt auch in deutschsprachiger Version:
aca_sig_setup_overwrite_de , GnuPG Signatur , ECDSA-Signatur

Das gleiche gilt für Windows 10 User – die zu ladenden Dateien sind unter Punkt C) zu finden.



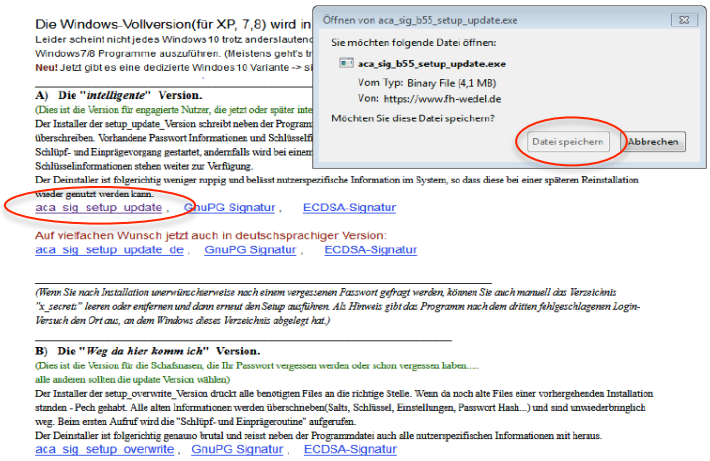
C) Speziell für Windows10:

aca_sig_update	GnuPG Signatur	ECDSA-Signatur
aca_sig_overwrite	GnuPG Signatur	ECDSA-Signatur

Bei Mehrbenutzer-Systemen, bei denen Sie nicht Administrator Privilegien haben gehen Sie wie folgt vor:
a) bitten Sie den Administrator, den Installer auszuführen. Dadurch werden die ausführbaren Programmdateien am richtigen Ort plaziert.
b) rufen Sie Anschließend noch einmal selbst mit Ihrem nicht-Administrator Account den Installer auf. Dadurch wird die aca_sig_b Datenstruktur mit x_secrets und key_tray Ordner angelegt. Der vergebliche Versuch, auch die Programmdateien zu schreiben, liefert ein paar Fehlermeldungen, die Sie ignorieren können.
Falls Sie selbst mit Administratorprivilegien den Installer ausgeführt hatten, das Programm aber mit nicht Administrator Privilegien nutzen möchten, ändern Sie die Zugriffsrechte auf die aca_sig_b Datenstruktur mit x_secrets und key_tray Ordner so, dass Sie als nicht Administrator dort Lese- und Schreibzugriffe ausführen können.
Ich werde versuchen, die Installation für diesen Fall im nächsten Update komfortabler zu gestalten.

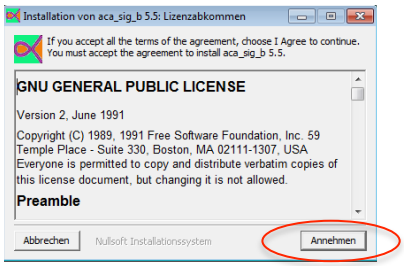
2. Nun klickt man entweder auf A) aca_sig_setup_update oder auf B) aca_sig_setup_overwrite

3. Bitte jetzt auf A) klicken und die Datei speichern - wir können uns das PW merken und nutzen somit die update Version.



4. Runtergeladene Datei öffnen – unter Downloads im jeweiligen Browser (Beispiel: Mozilla Firefox)

5. Lizenzabkommen „annehmen“



6. Zielverzeichnis auswählen – Speicherort des Programms

Die Windows-Vollversion(für XP, 7,8) wird in zwei Installationsvarianten angeboten:
Leider scheint nicht jedes Windows10 trotz anderslautender Behauptungen von Microsoft in der Lage zu sein, wxWidgets basierte Windows7/8 Programme auszuführen. (Meistens geht's trotzdem problemlos.)
Neu! Jetzt gibt es eine dedizierte Windoes10 Variante -> siehe unten.

A) Die "*intelligente*" Version.
(Dies ist die Version für engagierte Nutzer, die jetzt oder später intelligente Updates ausführen möchten)
Der Installer der setup_update_Version schreibt neben der Programmdatei nur Files auf die Festplatte, die keine nutzerspezifischen Daten überschreiben. Vorhandene Passwort Informationen und Schlüsselfiles werden respektiert und belassen. Nach dem Setup wird bei Erstinstallation der Schlüpf- und Einprägevorgang gestartet, andernfalls wird bei einem normalen Login das vertraute Passwort abgefragt und alle bisherigen Schlüsselinformationen stehen weiter zur Verfügung.
Der Deinstaller ist folgerichtig weniger ruppig und belässt nutzerspezifische Information im System, so dass diese bei einer späteren Reinstallation wieder genutzt werden kann.
aca_sig_setup_update , GnuPG Signatur , ECDSA-Signatur

Auf vielfachen Wunsch jetzt auch in deutschsprachiger Version:
aca_sig_setup_update_de , GnuPG Signatur , ECDSA-Signatur

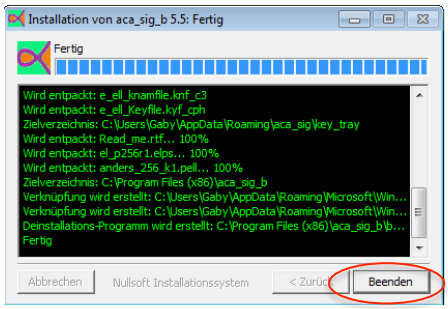
(Wenn Sie nach Installation unerwünschterweise nach einem vergessenen Passwort gefragt werden, können Sie auch manuell das Verzeichnis "x_secrets" leeren oder entfernen und dann erneut den Setup ausführen. Als Hinweis gibt das Programm nach dem dritten fehlgeschlagenen Login-Versuch den Ort aus, an dem Windows dieses Verzeichnis abgelegt hat.)

B) Die "*Weg da hier komm ich*" Version.
(Dies ist die Version für die Schafsnasen, die Ihr Passwort vergessen werden oder schon vergessen haben.....
alle anderen sollten die update Version wählen)
Der Installer der setup_overwrite_Version drückt alle benötigten Files an die richtige Stelle. Wenn da noch alte Files einer vorhergehenden Installation standen - Pech gehabt. Alle alten Informationen werden überschrieben(Salts, Schlüssel, Einstellungen, Passwort Hash...) und sind unwiederbringlich weg. Beim ersten Aufruf wird die "Schlüpf- und Einprägeroutine" aufgerufen.
Der Deinstaller ist folgerichtig genauso brutal und reisst neben der Programmdatei auch alle nutzerspezifischen Informationen mit heraus.
aca_sig_setup_overwrite , GnuPG Signatur , ECDSA-Signatur

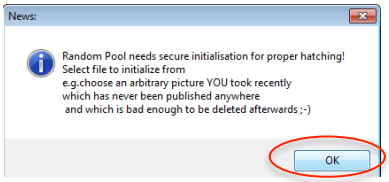
Auf vielfachen Wunsch jetzt auch in deutschsprachiger Version:
aca_sig_setup_overwrite_de , GnuPG Signatur , ECDSA-Signatur

---

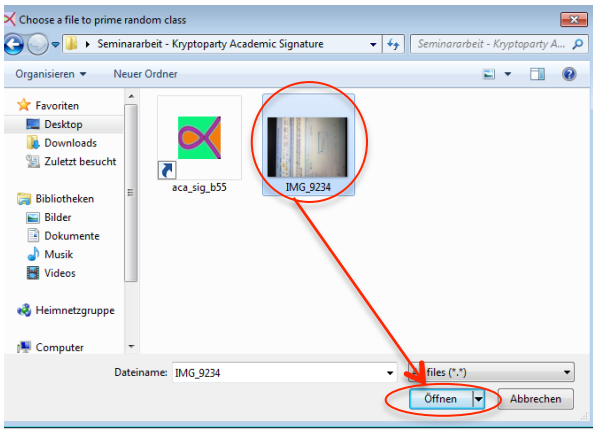
7. Nach Installation auf „Beenden“ klicken



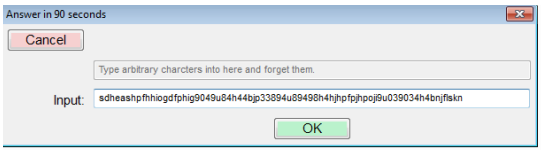
8. Starten der Exe.datei von Academic Signature durch Doppelklick auf das Icon. Nun wird man nach einem random Bild gefragt, dass später wieder gelöscht werden kann



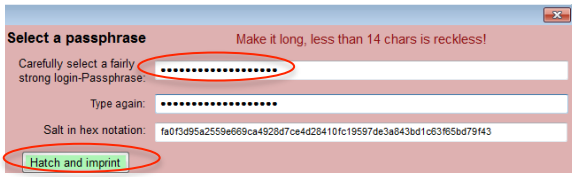
9. Nun wird man nach einem random Bild gefragt, dass später wieder gelöscht werden kann



10. Nun wird ein weiterer Schutz angefordert: Um zu verhindern, dass ein Angreifer auf ihre Datei zugreift, die Sie ausgewählt haben, wird eine möglichst lange Zeichenfolge eingegeben. (Diese Zeichenfolge muss man nicht behalten)

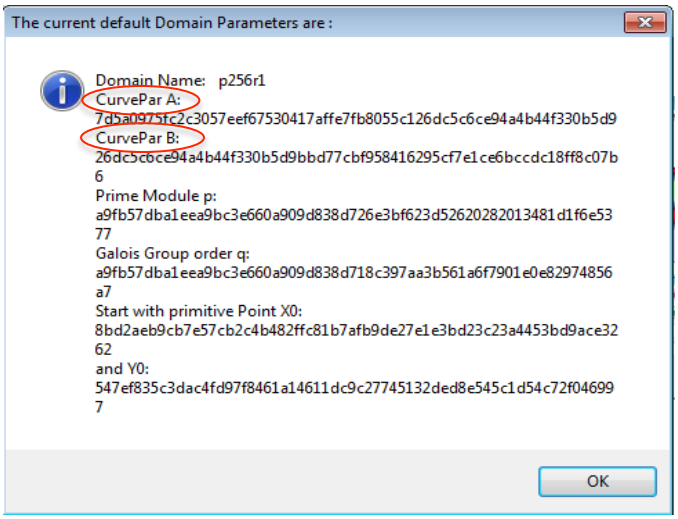


11. Passwort wählen im nächsten Schritt (mind. 14 Buchstaben!) merken oder notieren: 1234567890123456789
Dann klicken Sie auf „**Hatch and imprint**“



7. ZUSAMMENHANG – ELLIPTIC CURVES & ACADEMIC SIGNATURE

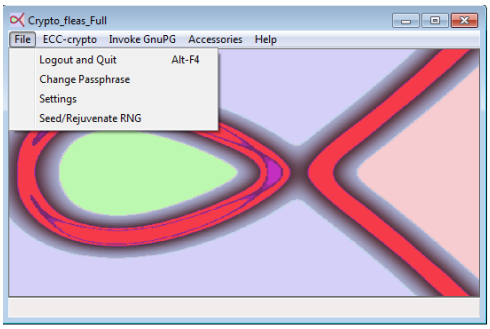
Bevor wir mit den grundlegenden Funktionen beginnen, hier einmal der Zusammenhang zu den elliptischen Kurven. In der folgenden Grafik sind die Parameter A und B der Funktion der elliptischen Kurve ersichtlich.

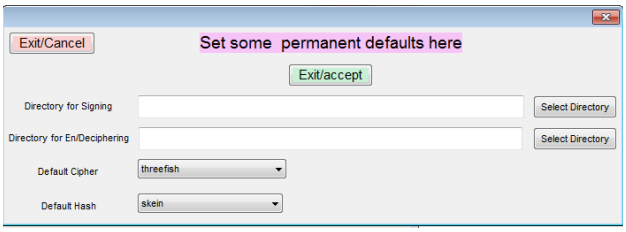


8. ALLGEMEINE INFORMATIONEN

FILE

a. **Logout and Quit:**
   Button zum Ausloggen
b. **Change Passphrase:**
   Passwortänderung möglich
c. **Settings (für permanente Einstellungen):**
   i. „Directory for signing“:
   →hier kann das Verzeichnis der signierten Dateien verändert werden
   ii. „Directory for En/Deciphering“:
   →hier kann das Verzeichnis der verschlüsselten/ entschlüsselten Dateien geändert werden
   iii. „default“:
   →hier kann ein Verzeichnis als Standard-Verzeichnis gewählt werden (z.B. threefish – Blockverschlüsselung)

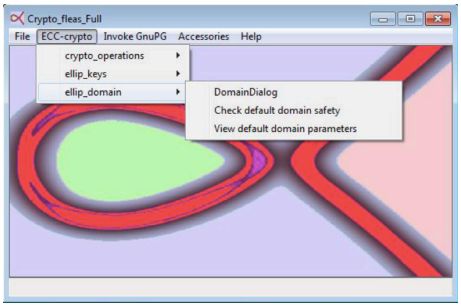




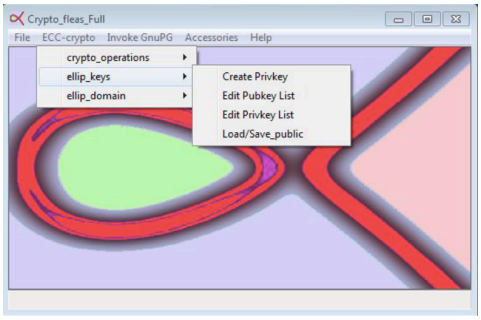
d. **Seed/ Rejuvenate RNG:**
   Verjüngen der Sicherheit der ausgewählten Datei
   →hierbei wird der Zufallsgenerator neu mischen

ECC-CRYPTO (ELLEPTICAL CURVE CRYPTO):

a. **ellip_domain:**
   →Domain laden
   →Parameter der Standarddomäne anzeigen lassen
   →Sicherheit der Standarddomäne prüfen
   →Domäne bearbeiten, einstellen



b. **ellip_keys:**
   →Private Key erstellen, laden, speichern
   →Public Key laden, speichern
   →Public Key-Liste kann hier überarbeitet werden
   →Lösche/ Importiere Private Key (von einer Datei)



c. **crypto_operations:**

Operationen durchführen, wie zum Beispiel

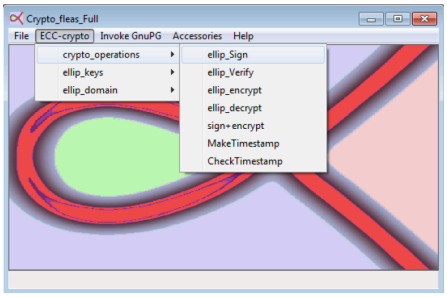
→Datei signieren

→Datei verifizieren

→Datei verschlüsseln/ entschlüsseln

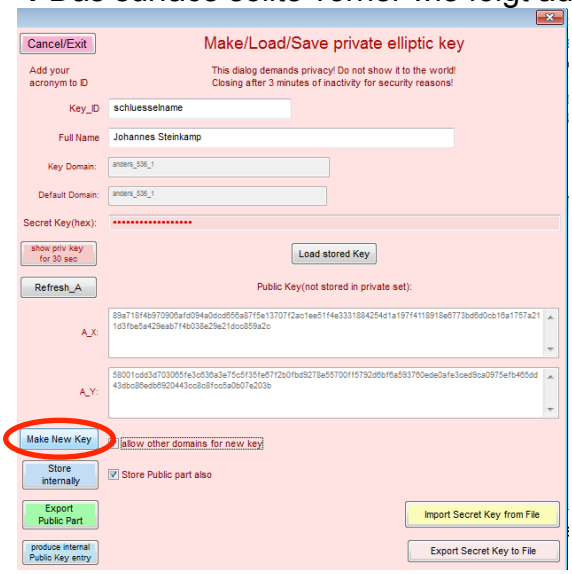
→Datei gleichzeitig signieren und verschlüsseln

→Zeitstempel erstellen und prüfen

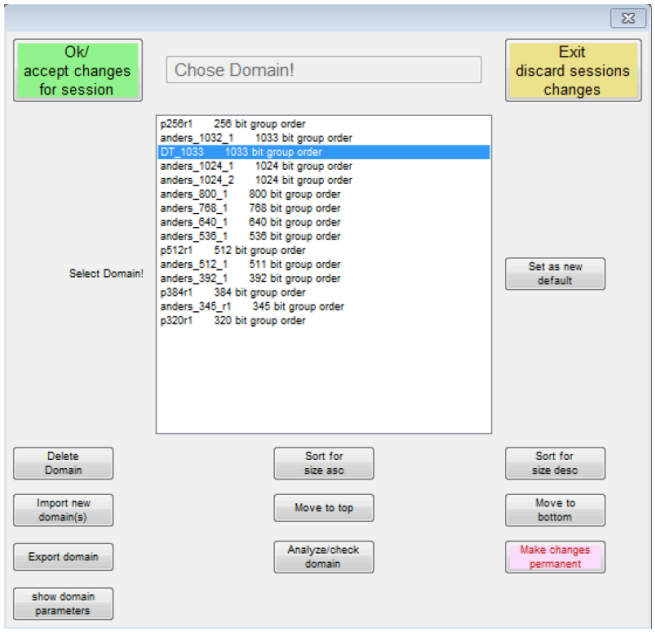


9. CREATE PRIVATE/ PUBLIC KEY

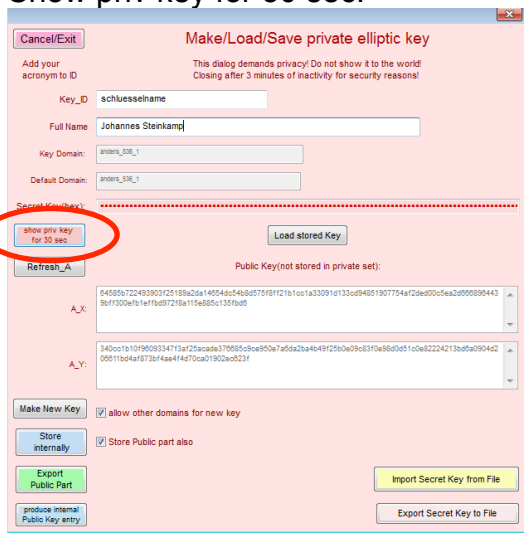
1. ECC-crypto → ellip_keys →create/ load/ save_priv
2. Key_ID – geben Sie ihrem Key eine Benennung/ einen Namen
3. Full Name – geben Sie ihren Namen ein
4. Haken setzen bei „*allow other domains for new key*“ – dies bewirkt, dass eine neue Domain für den Key ausgewählt wird
5. Drücken Sie „**Make New Key**“
   →Das surface sollte vorher wie folgt aussehen:

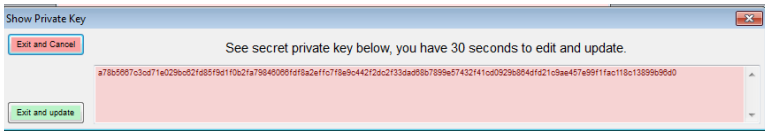


6. Nun öffnet sich ein neues Fenster – hier kann man eine Domäne wählen
   Info: ab 256 Bit gilt ein Schlüssel als sicher gegenüber BruteForce Angriffen für symmetrische Verfahren.
   Ich wähle hier nun den DT_1033 Schlüssel, der der aktuellen Situation entsprechend Donald Trump gewidmet ist. Die Zeit des Verschlüsselns und Entschlüsselns ist abhängig von der Schlüssellänge (z.B. 1033 bit).



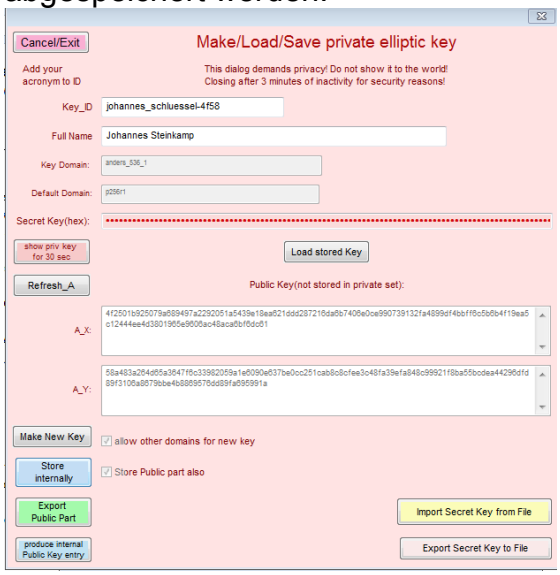
7. Hier kann man sich den privaten Schlüssel für 30 Sekunden anzeigen lassen. Show priv key for 30 sec:



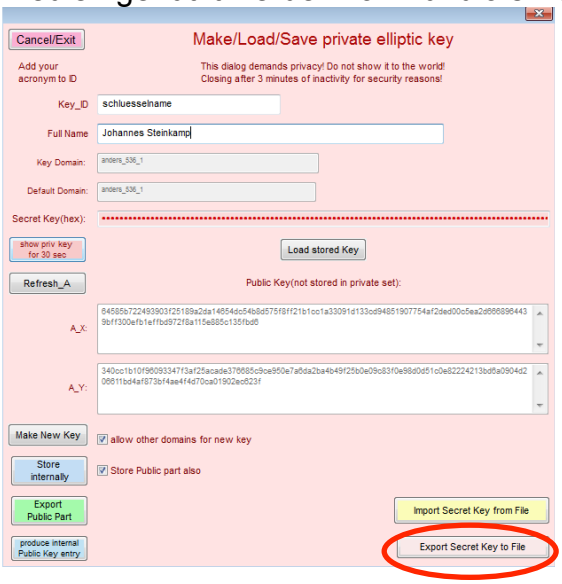


8. Nun wird der Key gespeichert, indem man den Button „**Store Internally**“ aktiviert – falls ein neuer Schlüssel gewählt wurde, muss nun noch einmal die „*Key_ID*“ und der „*Full Name*“ eingegeben werden.

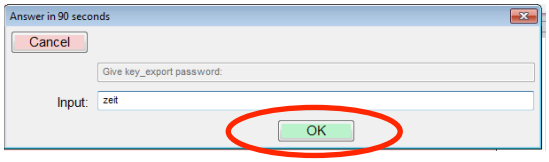
9. Durch aktivieren des Buttons „**Export Public Part**“ kann nun noch der öffentliche Schlüssel als Datei in einem Ordner als .pell-Dateityp abgespeichert werden.



10. Als nächster Schritt wird durch „**Export Secret Key to File**“ der geheime Schlüssel als Datei exportiert, damit der private Schlüssel auch auf anderen Medien genutzt werden kann und als Datei verwaltbar und verwendbar ist.



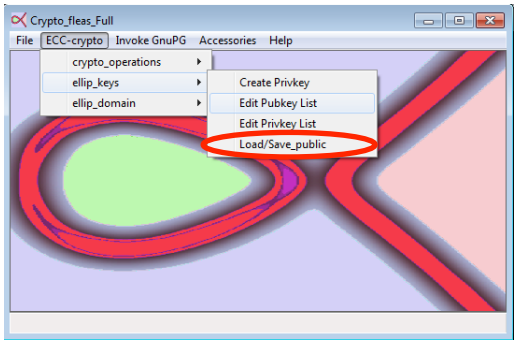
11. Anschließend öffnet sich dieses neue Fenster, um ein neues Passwort für diese Datei zu erstellen
→Input: „zeit“
→auf „Ok“ klicken
→Speicherort des Schlüssels: Key_tray oder der Speicherort, wo das Programm aca_sig gespeichert wurde



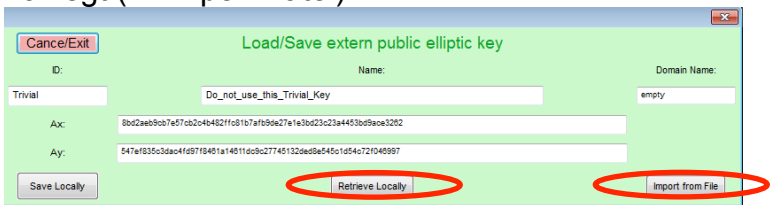
12. Nun wurde ein öffentlicher (public part) und ein privater (secret key) Schlüssel erstellt und gespeichert.

10. PUBLIC KEY IN GESCHÜTZTEN BEREICH PORTIEREN

1. ECC-crypto → ellip_keys →Load/Save_public

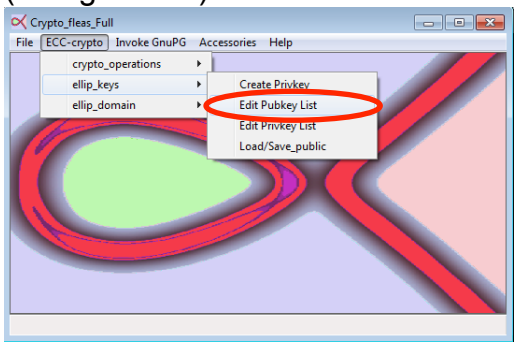


2. Öffentlichen Schlüssel, der geschützt werden soll, hochladen durch "**Retrieve Locally**" – ich wähle hier nun meinen erstellten Schlüssel aus. Falls der gewählte öffentliche Schlüssel schon in der „pubkey-Liste" vorhanden ist, gibt es eine Meldung.
   Auf „**Import from File**" kann ein Schlüssel importiert werden, der als Datei vorliegt (z.B. .pell-Datei).



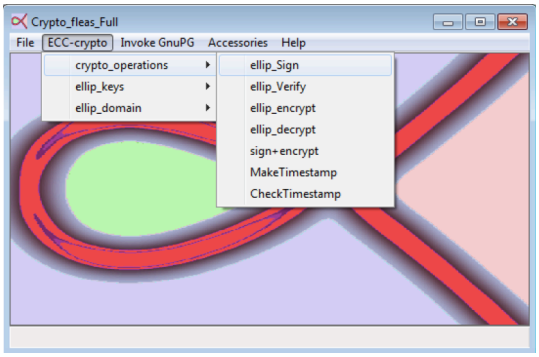
3. Nun wird auf „**Save Locally**" der ausgewählte Schlüssel abgespeichert

4. Der Schlüssel ist nun in der pubkey Liste vorhanden und kann dort verwaltet (z.B. gelöscht) werden.



11. DOKUMENT ELEKTRONISCH SIGNIEREN

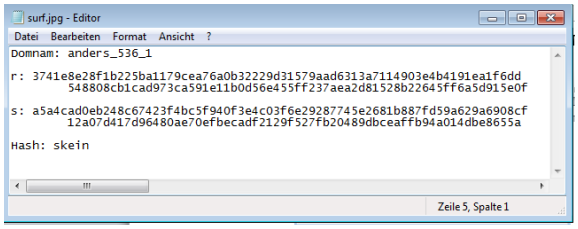
1. Crypto_operations →ellip_Sign



2. Unter „**select file**“ kann nun eine Datei ausgewählt werden, die signiert werden soll.

Nun kann noch ein Hash ausgewählt werden (z.B. Skein (1024 bit)).

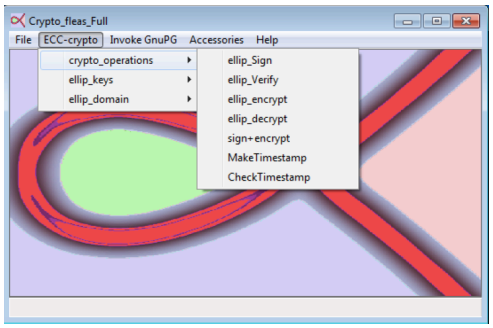
3. Die Signatur kann nun als Textdatei (mit Hilfe des Editors) angeschaut werden:



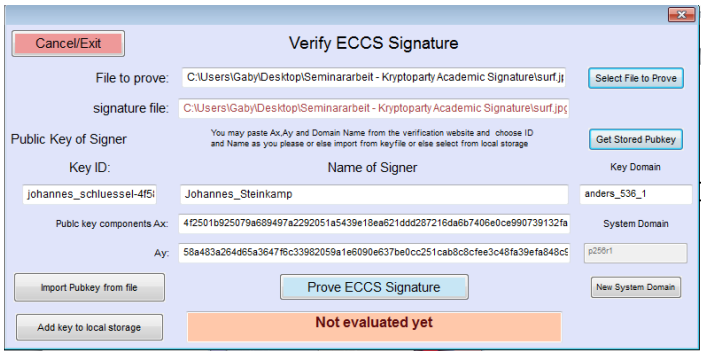
→Domainname
→r, s – 2 Parameter
→der genutzte Hash

12. VERIFIKATION EINER ELEKTRONISCHEN SIGNATUR

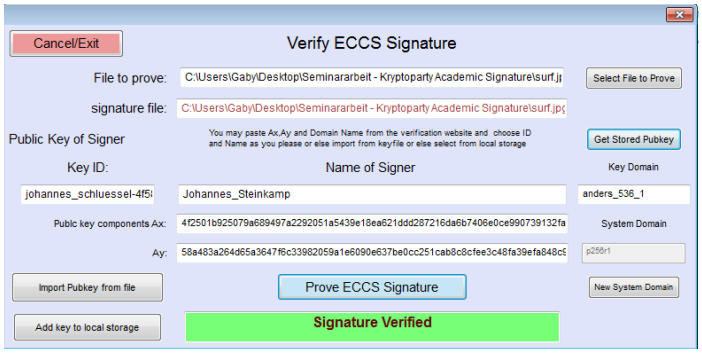
1. Crypto_operations →ellip_verify
   Es soll hier nun geprüft werden, ob die Datei mit der Signatur und dem öffentlichen Schlüssel zusammenpasst.



2. →„**Select File to Prove**“ - Datei hochladen, die geprüft werden soll (z.B. das soeben signierte Bild(.jpg))
   →”**Get stored Pubkey**“ - öffentlichen Schlüssel laden/ benutzen, der der Person, die die Signatur erstellt hat, gehört
   Es besteht auch wieder die Möglichkeit einen Pubkey from file zu importieren

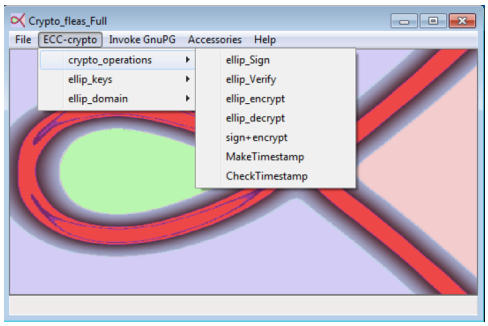


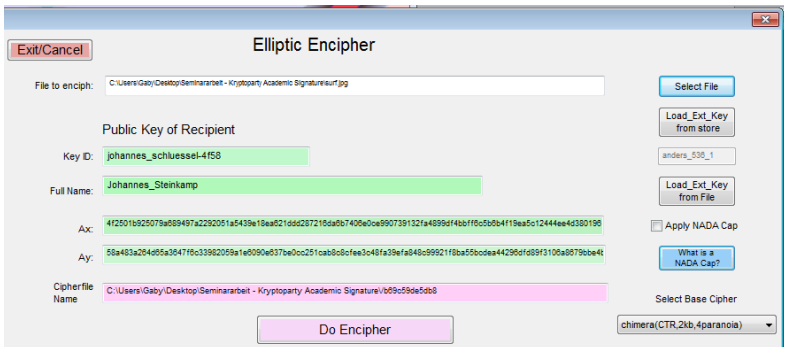
3. →“**Prove ECCS Signature**“



13. CHIFFRE – VERSCHLÜSSELN, ANZEIGEN, ENTSCHLÜSSELN

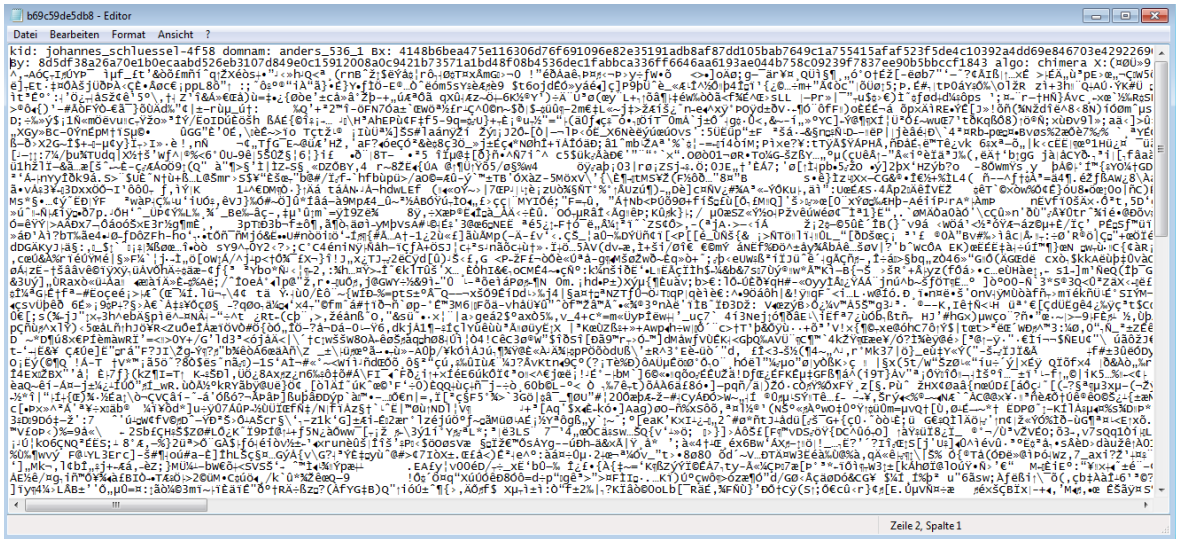
1. Chiffre erstellen
   Crypto_operations → Ellip_encrypt





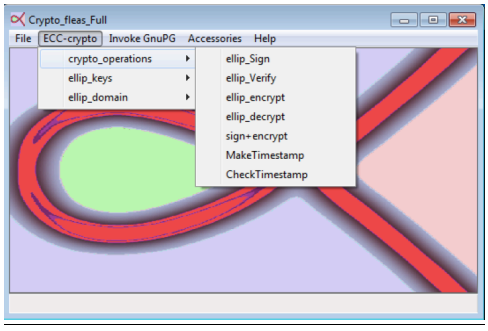
→Zuerst wählen Sie eine Datei aus, die chiffriert werden soll
→Load_Ext_Key from store → hier wird mit der öffentliche Schlüssel gewählt und dann kann die Datei nur mit dem privaten Schlüssel dechiffriert werden
→“Apply NADA Cap“ bedeutet, dass die Key_ID nicht angezeigt wird und somit muss derjenige, der die Datei dechiffrieren will, den öffentlichen Schlüssel erraten müsste
→“Cipherfile Name“ ist ein Dateiname, der mit zufälligen Zahlen versehen ist – dies ist eine zusätzliche Sicherheit, da derjenige, der die Datei abfangen möchte, keine Kenntnis über den Dateinamen erlangen kann.
→“Select Base Cipher“ – wählen Sie einen beliebigen Algorithmus
→“Do Encipher“ – Chiffre erzeugen

2. Erstellte Chiffre im Editor



→Kid: öffentlicher Schlüssel
→domnam: Domänenname
Wenn wir bei NADA einen Haken gesetzt hätten, würden uns diese Informationen vorenthalten bleiben
→BX, BY – zwei Koordinaten
→algo: chimera – der gewählte Algorithmus

3. Chiffre entschlüsseln
Crypto_operations → Ellip_decrypt



→„Select File“ – Datei, die dechiffriert werden soll, wählen
→“Select privKey from Store“ – geheimen Schlüssel wählen, um zu dechiffrieren
→“Do Decipher“ – Dechiffrieren durchführen