## Some Screenshots of Academic Signature

Academic Signature is dialog oriented and contains more than 20 different dialog windows. To give you an overview of what is to be expected during usage, I took some shots of core dialogs. The shots were taken from version b17.

•		÷Χ
Cancel/Exit	ECCS Sign File	
File to Sign:	/media/200C-D29F/Klausuren/feb11/An_Anmeld_Klausur.ods	Select File
ECCS-Key ID:	anders_256_k1	Select other Key
Full Name:	Prof.Dr.Michael Anders	Select Hash
Current Domain / Key Domain must match	Michi dustemann	Fleas_x2(2048 bit)
p256r1	Create Signature	
p256r1		

Illustration 1: ECDSA Signing Dialog. You select the file to be signed, the private key to be used and select which hash algorithm is to be used. Clicking "Create Signature" will produce the signature file sharing the path and name with the document file with the additional extension "ecsg". Note that the red background color indicates an operation involving the private key.

	•	- *	$\uparrow$ $\times$	
	Cancel/Exit	Verifiy ECCS Signature		
1	File to prove:	/home/micha137/downloads/aca_sig/aca_sig_b17_3	Select File to Prove	
2	signature file:	/home/micha137/downloads/aca_sig/aca_sig_b17_3.ecsg	)	
[	You may paste Ax,Ay and Domain Name from the verification website and choose ID and Name as you please			Verified!
1	or else import from keyfile or else select from local storage	Public Key of Signer	Get Stored Pubkey	Holder of public Key signed this file with corresponding private Key.
	Key ID:	Name of Signer	Key Domain Name	
1	anders_256_k1	Prof.Dr.Michael_Anders	p256r1	
4	Publc key components Ax:	2f3ce66b514a599b5318e057e4c2558fe27fcf6d6d86fb204e4a1d08224f9bc8	Current Domain name	Ok
	Ay:	8f6e9a6921241d1e00d84d35baf3fbf0b62d35cb74441e6798047ac8d228d36f	p256r1	
-	Import Pubkey from file	Prove ECCS Signature	Load other Domain	
ł	Add key to local storage	Signature Verified		

Illustration 2: Shot of the Verification Dialog after a successful signature proof. You select which file to prove and the public key to prove against. The public key can be set from internal protected storage(Button"Get Stored Pubkey"), from importing a human readable \*.pell file containing the key(Button "Import Pubkey from File") or can be simply pasted into the Text controls from any textual source in hex-notation. Upon signature acceptance, the green message appears at the bottom and an additional message box appears.

••	+ ×
Cancel/Exit	Make/Load/Save private elliptic key
	This dialog demands privacy! Do not show it to the world! Closing after 3 minutes of inactivity for security reasons!
Key_ID	BSE
Full Name	Mad Cow Desease
Domain Name:	p256r1
Current Domain:	p256r1
Secret Key(hex):	[]
show priv key for 30 sec	Load stored Key
Refresh_A	Public Key(not stored in private set):
A_X:	95423d51e0f208ff2571e927382b0dcd822174ae1d40ae33f23aac949eef3239
A_Y:	6ee1fc4336e1f7b7e82a5b4526a62841769145697563559df42a4c539e3ea734
Make New Key	Store internally
Import Secret Key from File	Export Public Part
Export Secret Key to File	

Illustration 3: Dialog for handling private key information.

Procedures and associated secondary dialogs accessible from this dialog are e.g. retrieve private key from internal storage("Load stored Key"), produce new private public key pair("Make New Key"), export public part to file, import private key from enciphered file or export private key to an enciphered file. Since this dialog has access to your personal "Fort Knox" of secrets it will close automatically if unattended for some time. Hinting to involvement of private keys, the background color is reddish.

(	••		+ ×
	Cance/Exit	Load/Save extern public elliptic key	
	ID:	Name:	Domain Name:
1	anders_256_k1	Prof.Dr.Michael_Anders	p256r1
1	Ax:	2f3ce66b514a599b5318e057e4c2558fe27fcf6d6d86fb204e4a1d08224f9bc8	
	Ау:	8f6e9a6921241d1e00d84d35baf3fbf0b62d35cb74441e6798047ac8d228d36f	
	Save Locally	Retrieve Locally	Import from File

Illustration 4: Dialog for dealing with and selecting public key information.

Again the public key can be set from internal protected storage(Button"Retrieve Locally"), from importing a human readable \*.pell file containing the key(Button "Import from File") or can be simply pasted into the Text controls from any textual source in hex-notation. The green background denotes that public keys are dealt with from here.

,	,		
Ì	••		$\bullet \times$
1	Exit/Cancel	Encipher and Authenticate	Do Sign _Enciph
1			select file
	File to encipher and Authenticate:	/home/micha137/downloads/aca_sig/aca_sig_t17_3	Select file
	output filename:	/home/micha137/downloads/aca_sig/752a6f4ff01b	
	Pubkey of Recipient:	cafe42 42	sel. key of recipient
1	Privkey of signer:	anders_256_k1 Prof.Dr.Michael Anders	sel. signing key
1	signing domain	current domain	recipient domain
	p256r1	p256r1	p256r1
		Domains must match!	
1	signing hash		encipher algor.
	Fleas_x5		Fleas_x5
-	hide recipient keyID and other info (better privacy) but recipient must guess!		

Illustration 5: Dialog for combined authentification and encryption. Use this dialog to first sign a file with your private key(red text control), merge signature and original document and encipher the resulting compound file for the intended recipient using his/her public key(green text control). You may select any file name for the cipher and can chose from a wide array of hash and symmetric enciphering algorithms. The public key of the messages target may or may not be disclosed in the cipher for either comfort or added privacy, respectively.

Upon successful deciphering, the recipient will be informed about the file being a compound of document and signature and the file will automatically split into separate document and signature files.