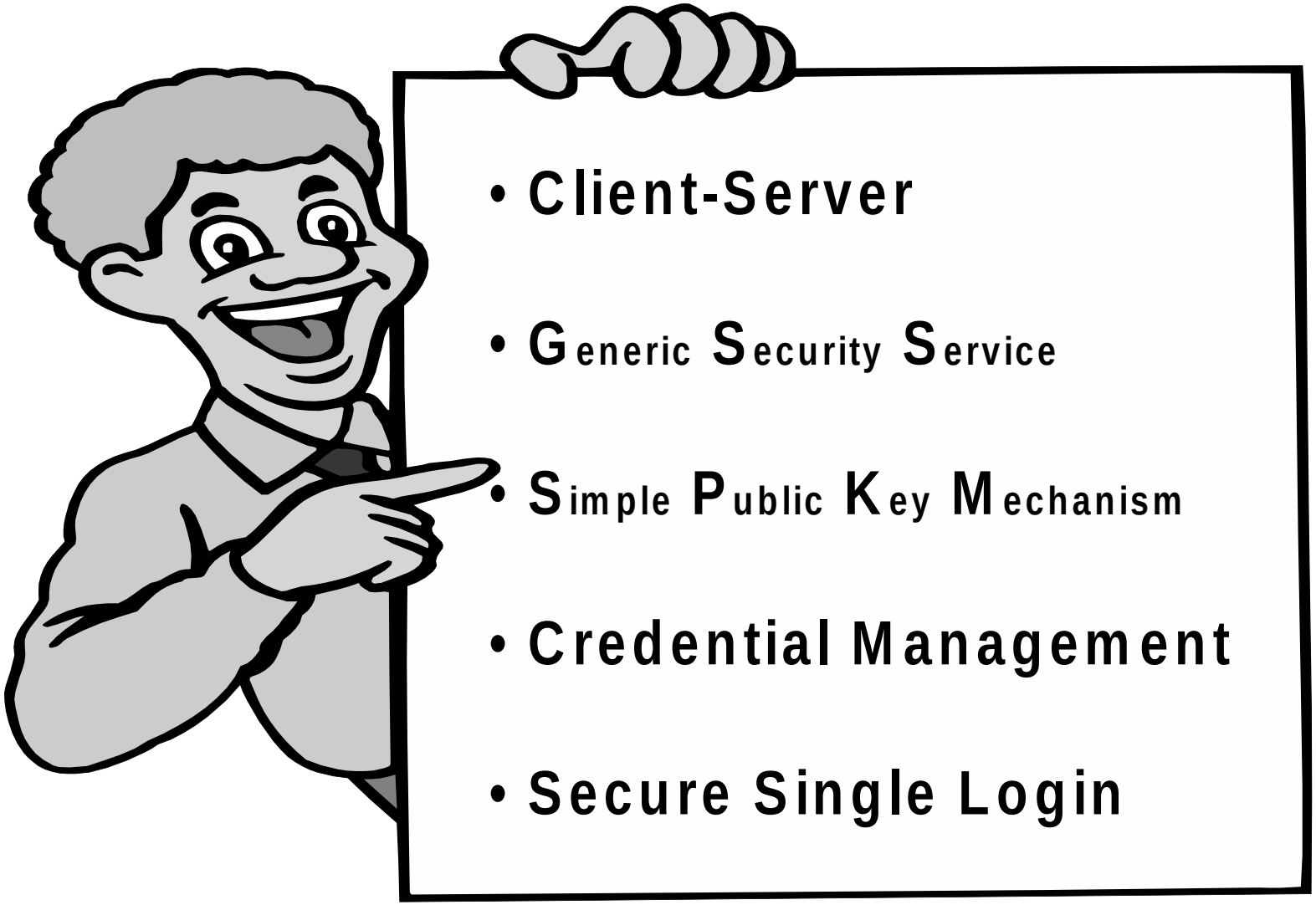
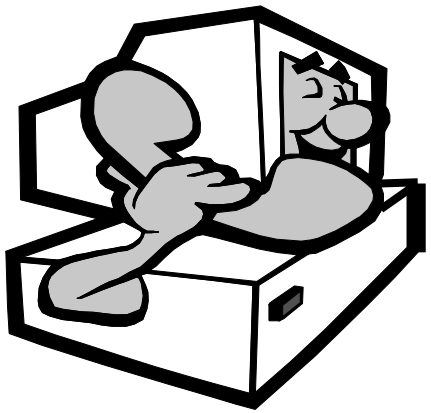
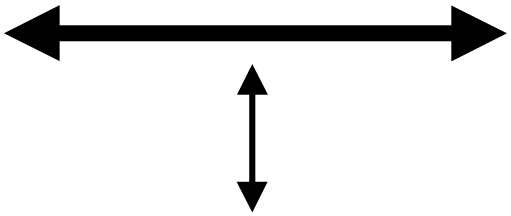


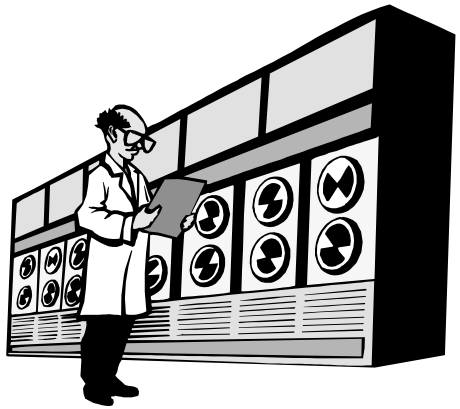
- Client-Server
- Generic Security Service
- Simple Public Key Mechanism
- Credential Management
- Secure Single Login

The Players



Client

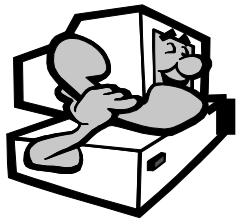




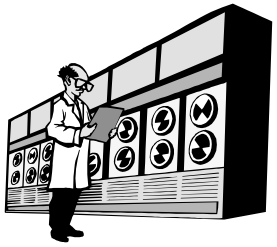
Server

Mr. BadGuy

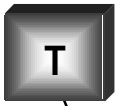
GSS



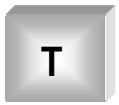
Client



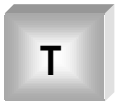
Server



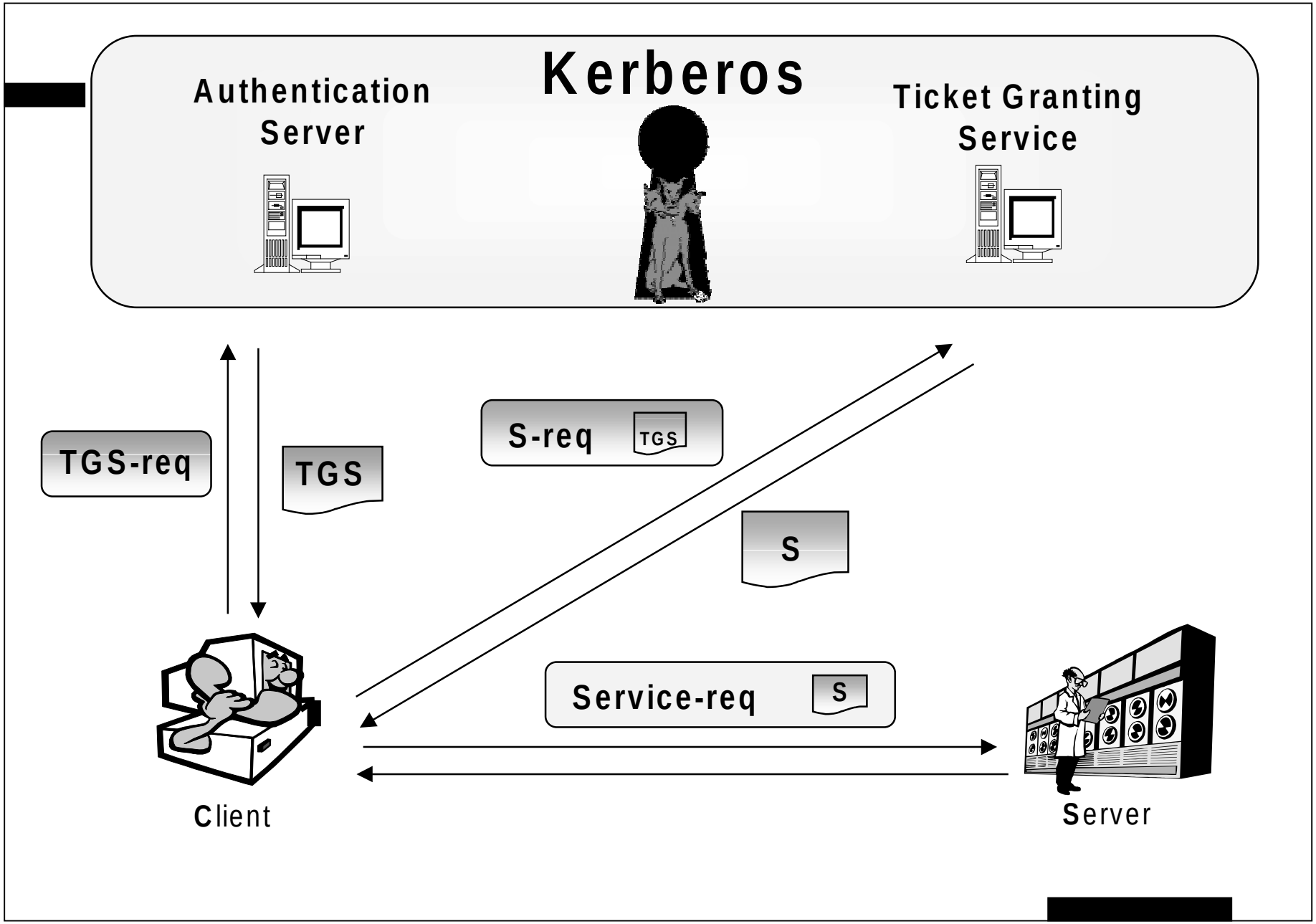
GSS_Init_sec_context
(..., $Cred_C$, S ,...)



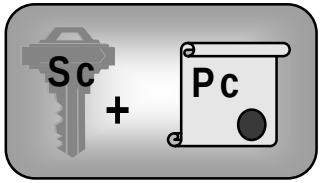
GSS_Accept_sec_context
(..., T , $Cred_S$)

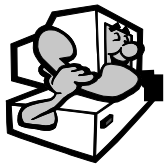


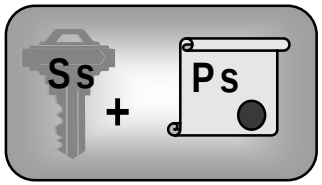
GSS_Init_sec_context
(..., T ,...)



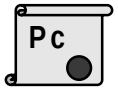
SPKM (3-Way-Auth)





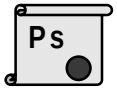


1. Rc, M=Rc|S|C, F=Sig(h(M),Sc)

2. M,F, 

3. Verify

4. Rs, Kcs, N=Rs|Rc|Kcs
G=Enc(N,Ps),
H=Sig(h(N),Ss)

5. G,H, 

6. Verify, decrypt G, I=Enc(Rs,Ps)

7. I

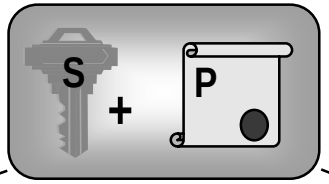
8. Rs'=Enc(I,Ss)=Rs?

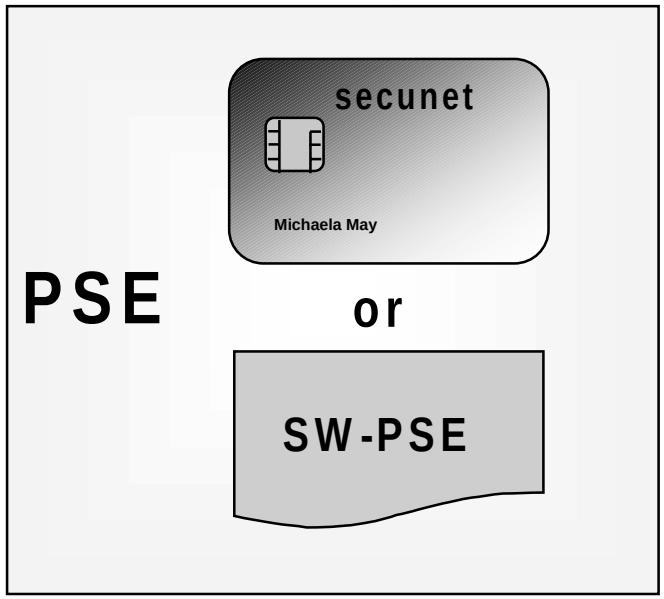
Credential Management for SPKM

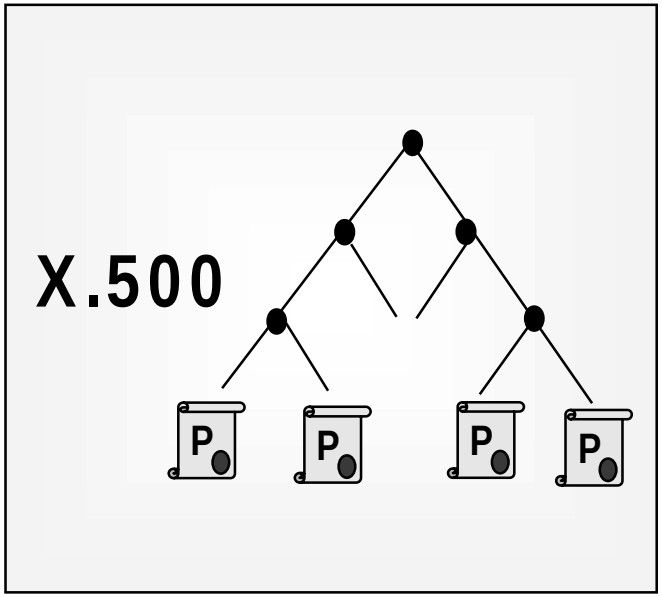
„The key management employed in SPKM is intended to be as compatible as possible with both X.509 and PEM, since these represent large communities of interest and show relative maturity in standards.“

SPKM

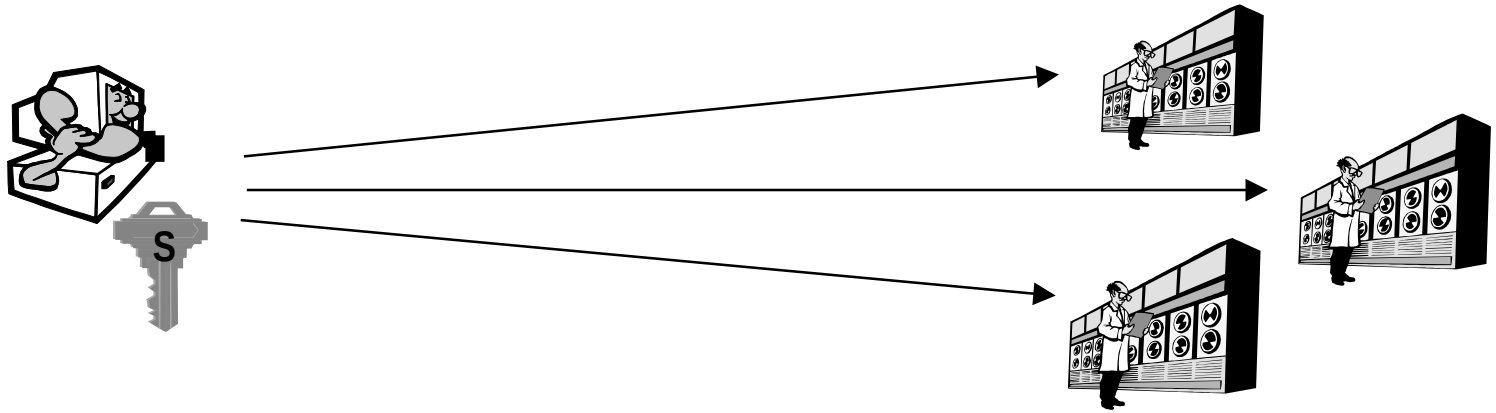
Credential Management







Multiple Connections

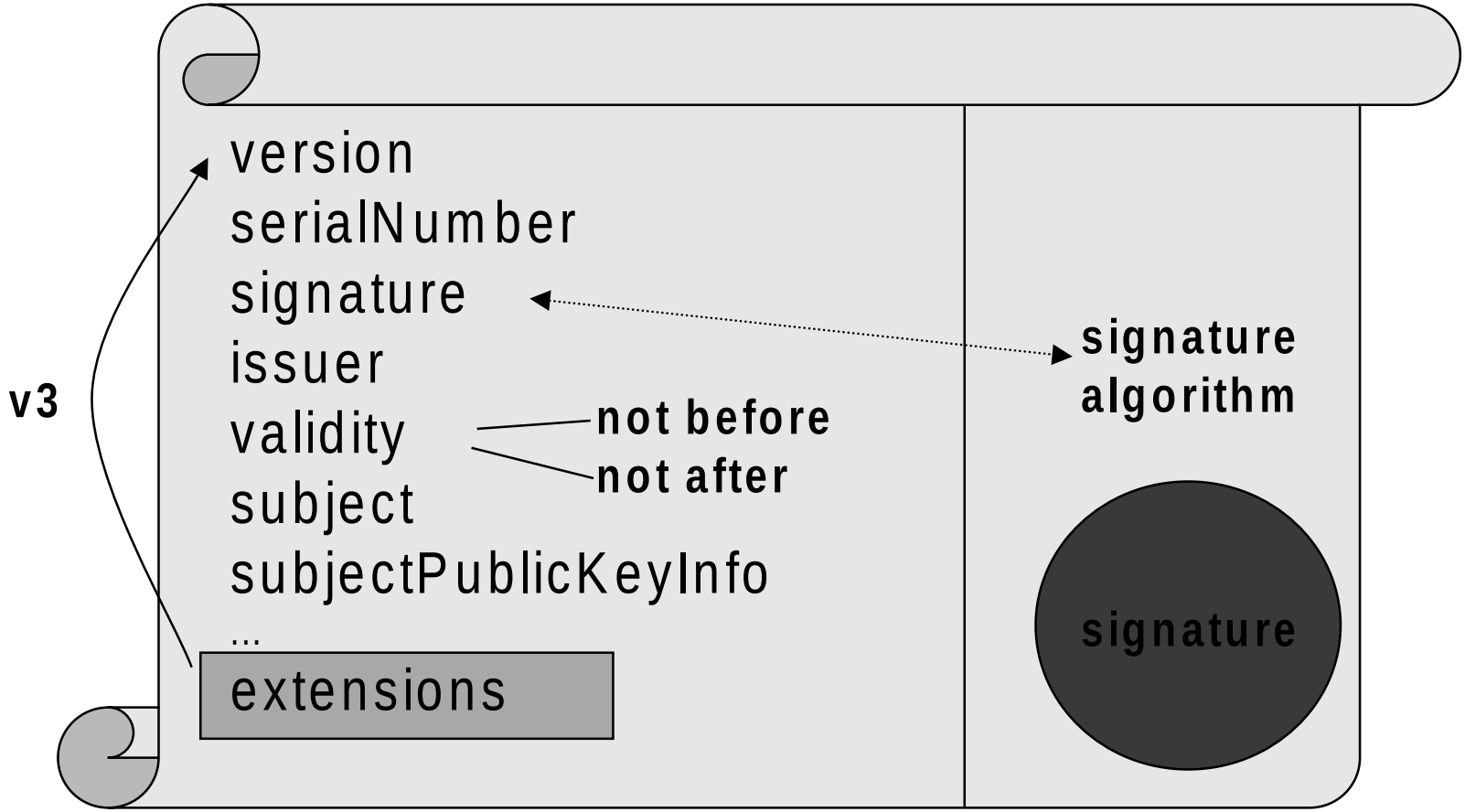


Credential Management	Usability	Security
Keep PSE accessible (for a long time)	☺	☹
Enter PIN to open PSE for every connection	☹	☺
Secure Single Login	☺	☺

Secure Single Login

Kerberos	SPKM
Get TGS-Ticket with <u>limited lifetime</u> to authenticate	Generate and (self) certify Public Key Pair with <u>limited lifetime</u> to authenticate

X.509 v3 - Certificates



X.509 v3 / PKIX - Extensions

Extension	
extnID	OID
critical	Boolean
extnValue	OctetString

subjectAltName
issuerAltName
basic Constraints

Boolean	cA
Integer	PathLenConstraint

Key usage

BitString	(0) digitalSignature
	(1) nonRepudiation
	(2) keyEncipherment
	(3) dataEncipherment
	(4) key Agreement
	(5) keyCertSign
	(6) cRLSign
	(7) encipherOnly
	(8) decipherOnly

Name Constraints

GenSubtree	permittedSubtrees
GenSubtree	excludedSubtrees

ExtendedKeyUsage

OID	KeyPurposeId

Examples:
id-kp-serverAuth
id-kp-clientAuth
id-kp-codeSigning
id-kp-emailProtection

Credential Management for SPKM

:=PKIX+incremental changes

new Key Purposes:
id-kp-SignTempCert
id-kp-Temporary

permanent

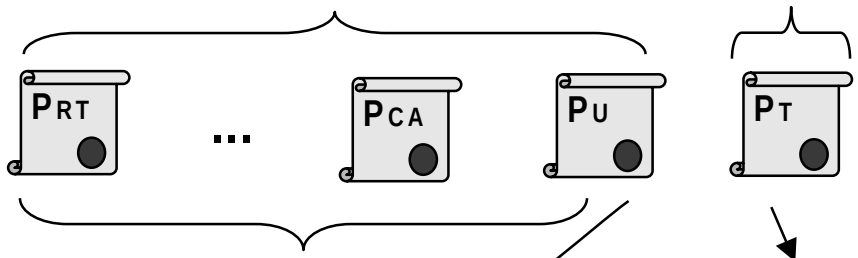
Issuer	CA
validity	u-notBefore
	u-notAfter
subject	User
subjectAltName	User-alt
issuerAltName	CA-alt
Keyusage	critical=TRUE
	digitalSignature
	nonRepudiation
ExtKeyUsage	critical=FALSE
	(id-kp-SignTempCert)
Basic Constraints	critical=TRUE
	cA=FALSE

temporary

Issuer	User
validity	t-notBefore
	t-notAfter
subject	User
subjectAltName	User-alt
issuerAltName	User-alt
Keyusage	critical=TRUE
	digitalSignature
ExtKeyUsage	critical=TRUE
	id-kp-Temporary
Basic Constraints	critical=TRUE
	cA=FALSE

Verification Procedure

1. PKIX conform — not present



2. PKIX conform

KeyUsage critical=TRUE
digitalSignature=TRUE
ExtKeyUsage (id-kp-SignTempCert)

Issuer = subject
issuerAlt= subjectAlt
validity.T-notBefore>validity.U-notBefore
validity.T-notAfter<validity.U-notAfter
KeyUsage critical=TRUE
nonRepudiation=FALSE
keyCertSign=FALSE
cRLSign=FALSE
ExtKeyUsage critical=TRUE
id-kp-Temporary is present
Basic Constraints critical=TRUE
cA=FALSE

Efficiency (Estimate)

	Security	Usab.	Time Efficiency (1024 Bit Mult.)			Space Efficiency (Byte)	
			Once	Session	Context	Secure	Insecure
Single Login	☹	☺	/	/	5526	/	/
Multiple Login	☺	☹	/	/	5526	/	/
SSLogin - RSA	☺	☺	/	108315	3758	/	740
SSLogin - DL (naive)	☺	☺	116000	1267	7236	20	1348
SSLogin - DL (prec.)	☺	☺	116517	675	6368	20	19780

... Comments appreciated

http://www.ietf.org/internet-drafts/draft-huehnlein-credman-spkm-00.txt

By Hans Schupp, GMD, Darmstadt, Germany
& me