Recent advances in Samba AD security

Douglas Bagnall

Catalyst IT & Samba Team

2026

New security features in Samba 4.24

and some from 4.23

security features ≠ security fixes

that’s another talk tomorrow

Domain encryption types changed to AES by default

following Microsoft CVE-2026-20833

for functional level 2008+

Domain encryption types changed to AES by default

default value for smb.conf option

`kdc default domain supported enctypes`

is now

`aes128-cts-hmac-sha1-96 aes256-cts-hmac-sha1-96`

(was: `arc4-hmac-md5` instead of aes128)

PKINIT improvements

“Windows Hello for Business” PKI mechanisms

key trust and *certificate trust*

key trust: like SSH
certificate trust: like smartcards, needs a certificate authority

Key trust is like SSH

private key kept safe, locally protected

public key copied to the right places

SSH: asymmetric keys used to create secure channel

Key trust: asymmetric keys used to obtain Kerberos TGT

Key trust requires 2016 schema

Certificate trust is like smart cards

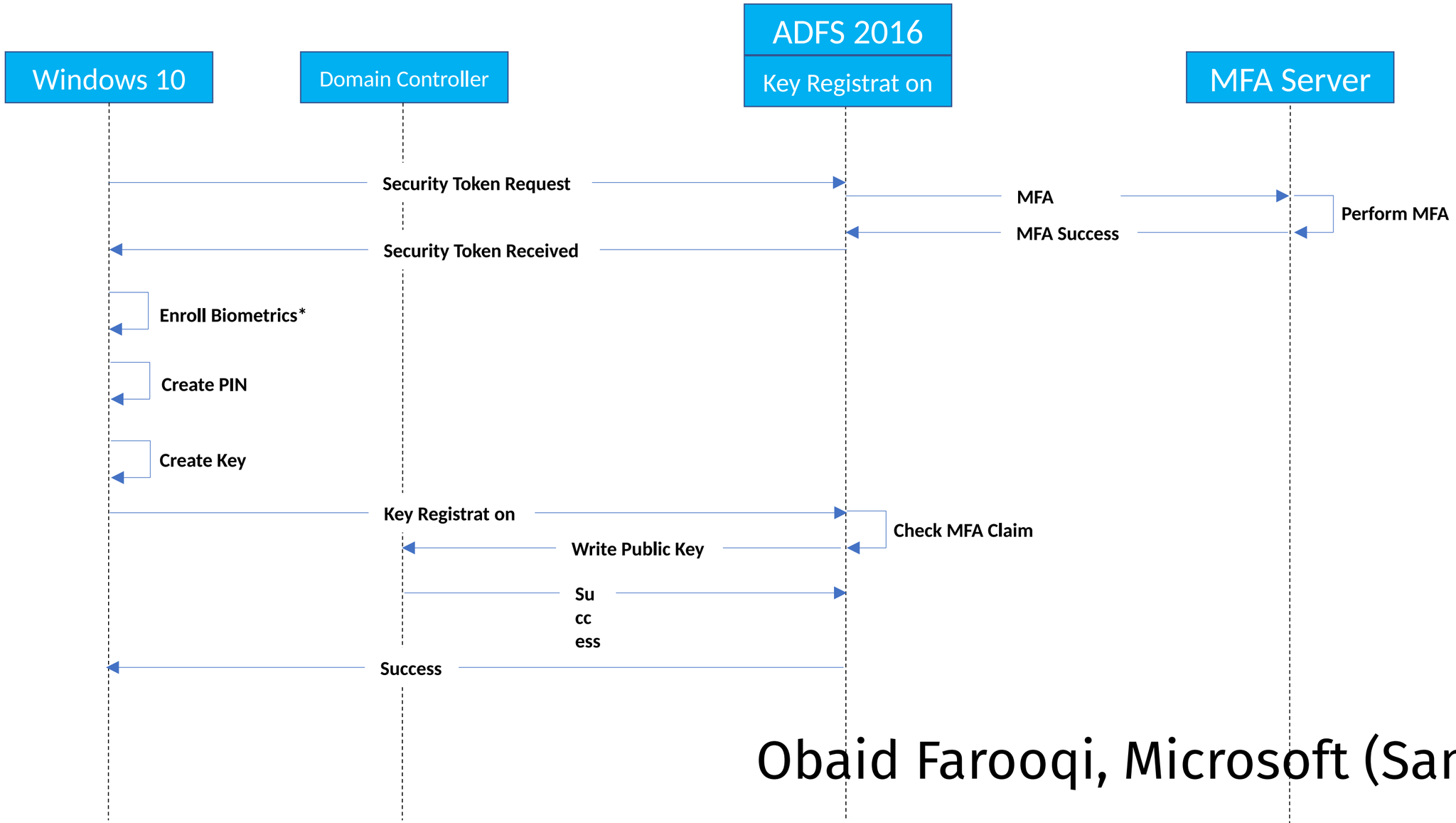
your laptop *is* the smartcard

keys are signed by centrally controlled authority

otherwise a lot like key-trust

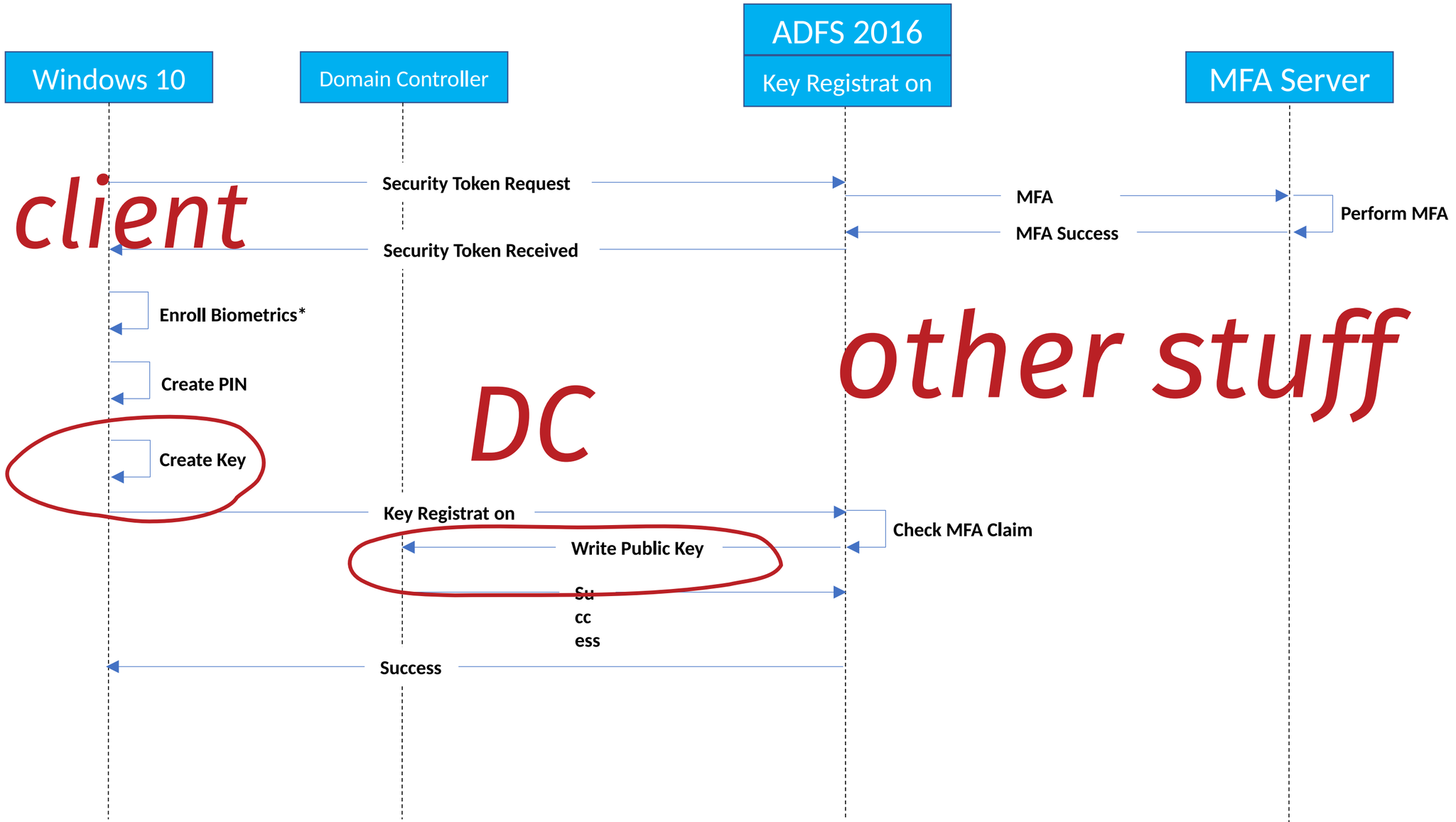
requires 2008R2 schema

Keytrust provision flow on Windows

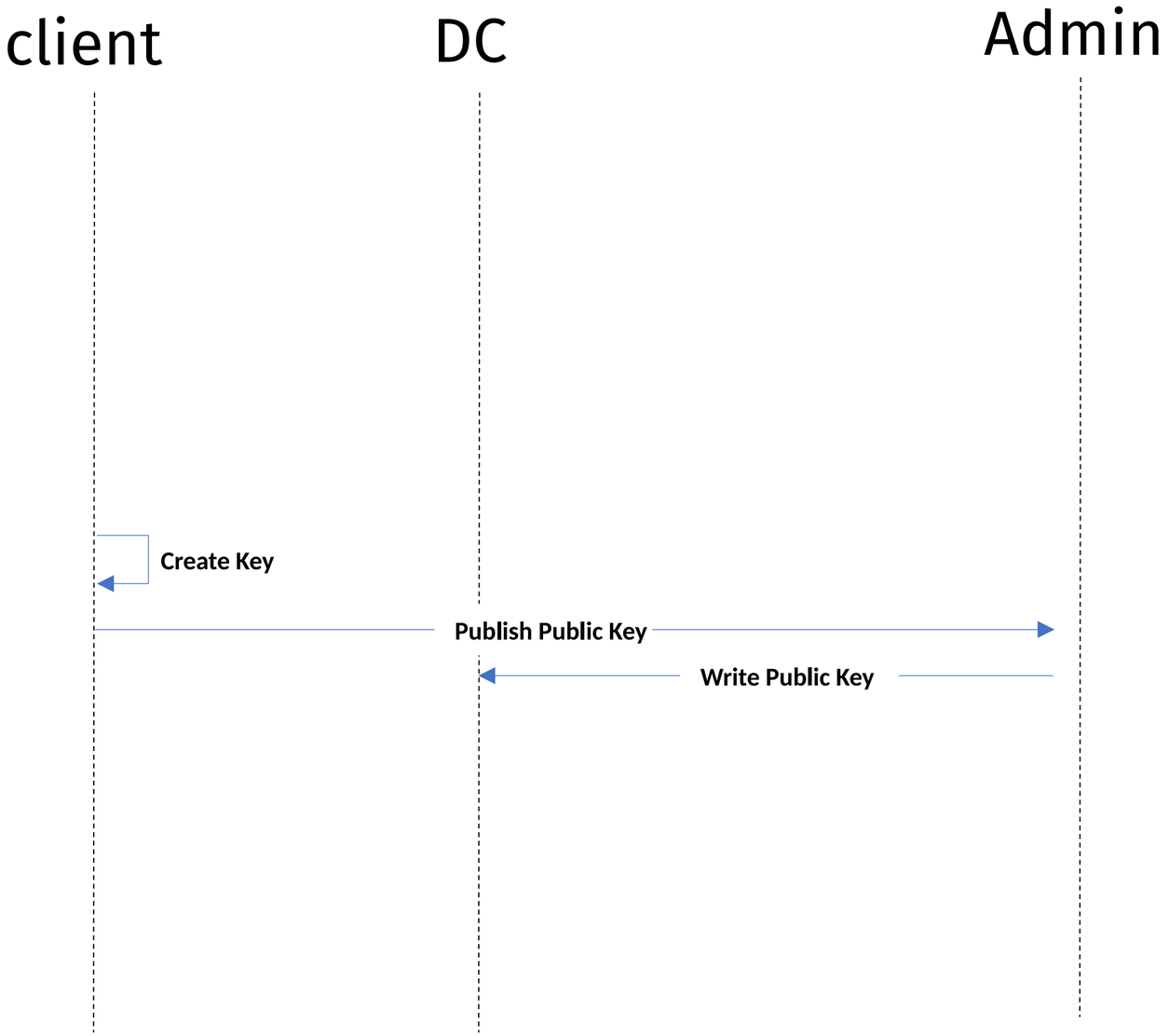


Obaid Farooqi, Microsoft (SambaXP 2019)

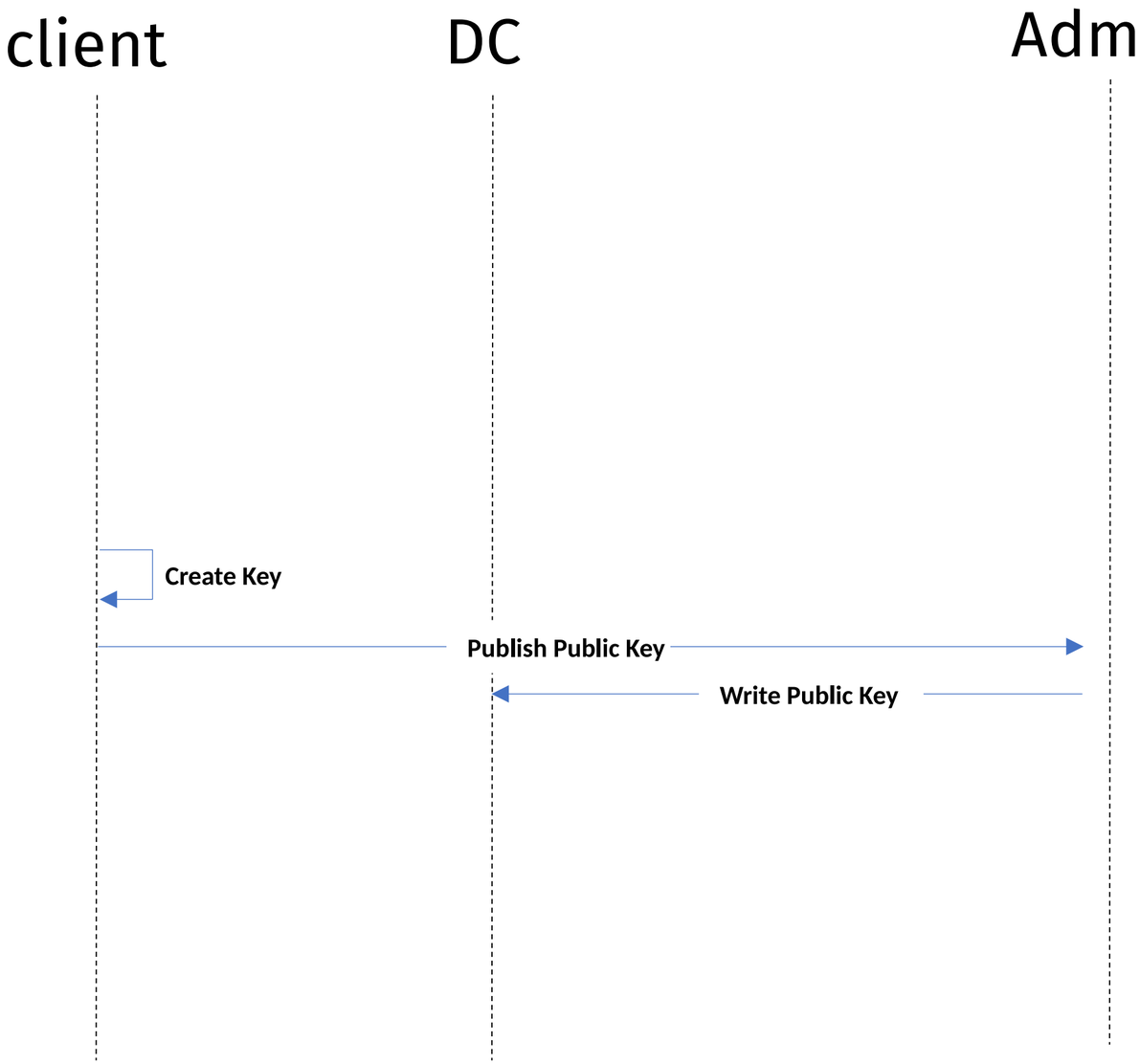
Keytrust provision flow on Windows



Keytrust provision flow with Samba



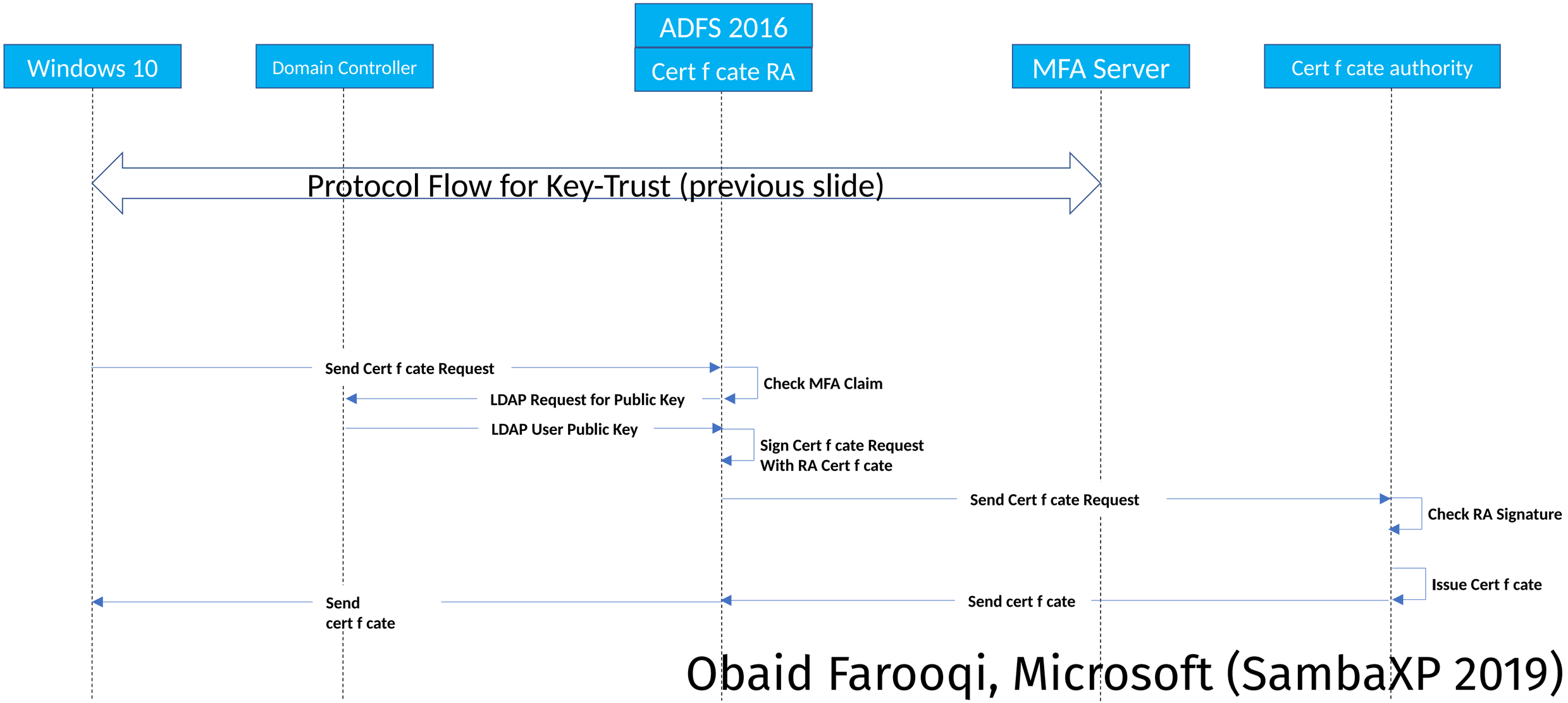
Keytrust provision flow with Samba



Administrator uses

```
samba-tool                  \
    {user,computer}         \
    keytrust                \
    add                     \
    $username               \
    $path_to_certificate.pem
```

Certificate provision flow with Windows



Obaid Farooqi, Microsoft (SambaXP 2019)

Certificate provision flow with Samba

Mapping certs to users

The DC could be confused about whose cert was whose

Microsoft KB5014754 defines “strong mappings” between objects and certificates

Defined in altSecurityIdentities attribute

Strong certificate mappings

Things that don't repeat and can't be spoofed:

- Certificate issuer and serial number
- Subject Key Identifiers (SKI)
- Certificate key SHA1

Not fully implemented:

- Certificate issuer and SID (with SID certificate extension)

strong mappings enforced by default

only for PKINIT certificate trust, which nobody uses with Samba

to disable enforcement:

```
strong certificate binding enforcement = none
```

disable enforcement when the user is older than the cert:

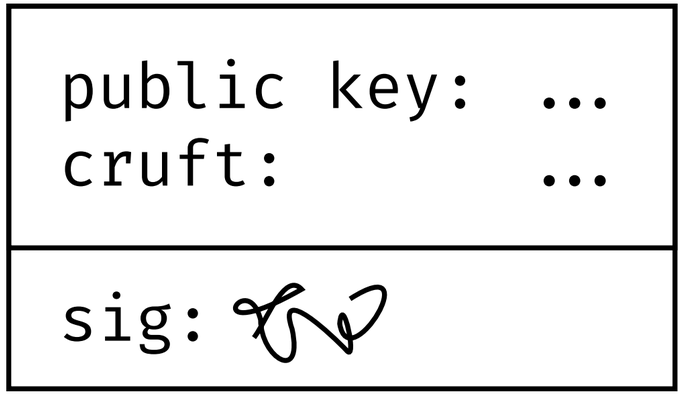
```
strong certificate binding enforcement = compatibility
```

this avoids many attacks.

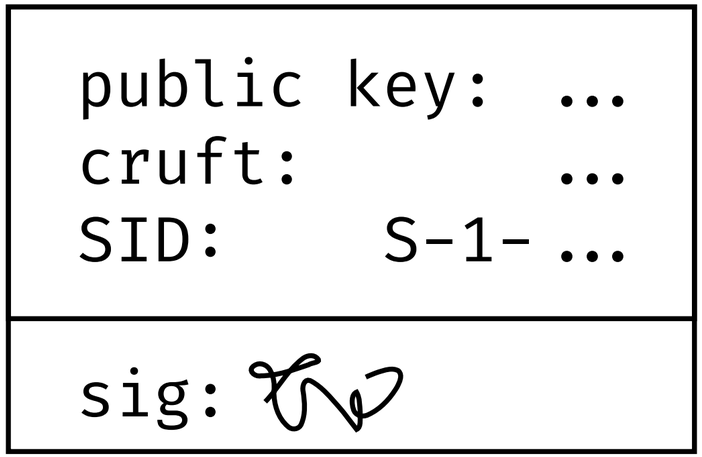
Certificate SID extension

so it knows which AD object it is for

Old cert:



New cert:



msDS-KeyCredentialLink

The attribute for storing public keys on objects

for key-trust think of msDS-KeyCredentialLink as ~/.ssh/authorized_keys

for certificate trust, it is there but other things matter more.

KDC can insist on canonicalisation

TGT must use the exact name in the request *unless* the client requests canonicalisation.

Allows the “dollar ticket attack” (bug 14785) because AD has loose mapping.

1. make AD machine account *Root$*
2. request ticket for *root* on unix domain member

KDC can insist on canonicalisation

Windows clients request canonicalisation, use FAST, expect *server* → *server$* mapping

Unix clients: perhaps none of these things

KDC can insist on canonicalisation

With

```
kdc require canonicalization = yes
```

a client TGT request will not succeed if it does not request canonicalisation

(returns *unknown principal* instead of *preauth required*)

safer matching without canonicalisation

With

```
kdc name match implicit dollar without canonicalization = no
```

The KDC will use:

- full AD name mapping for clients requesting canonicalisation
- no-$ mapping (but case-insensitive) for traditional clients

Which canonicalisation option to use

Clients are AD-aware:

```
kdc require canonicalization = yes
```

Some clients don’t set the canonicalise flag:

```
kdc name match implicit dollar without canonicalization = no
```

There is probably no situation in which you need the default behaviour

KDC always gives services canonical name

We don’t trust the TGT cname, so we issue service tickets with the canonicalised name from the PAC.

Unless you have:

```
krb5 acceptor report canonical client name = no
```

No difference if the client requested canonicalisation

KDC includes PAC by default

ignore client settings of `PA-PAC-REQUEST` unless

```
kdc always generate pac = no
```

You *might* not want the PAC if

- clients are old and confused
- accounts have very many group memberships

Questions?

douglas.bagnall@catalyst.net.nz