RSAConference2019

San Francisco | March 4–8 | Moscone Center

BETTER.

SESSION ID: IDY-W03

Understand Credential Security: Important Things You Need to Know About Storing Your Identity

Paula Januszkiewicz

CQURE: CEO, Penetration Tester / Security Expert

CQURE Academy: Trainer

MVP: Enterprise Security, MCT

Microsoft Regional Director

Contact: paula@cqure.us | http://cqure.us

Featured TechEd 2012 Speakers

More featured speakers →

Wally Mead

John Craddock

Mark Russinovich

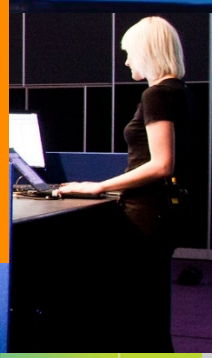
Paula Januszkiewicz

Microsoft

CQURE ✕ ACADEMY©

We are proud to announce that
Paula Januszkiewicz
was rated as
No 1 Speaker
at Microsoft **Ignite!!!**

May 4-8, 2015
Chicago, IL



TechEd
Europe 2013

Learn. Connect. Explore.
Join us in Madrid 25–28 June

Brad Anderson

Jon DeVaan

Mark Russinovich

Brian Keller

Paula Januszkiewicz

Mark Minasi

Jeffrey Snover

John Craddock

Scott Woodgate

Marcus Murray

RSACONFERENCE
2011 信息安全国际论坛
International INFOSEC Forum
CHINA 2011

Where The World Talks Security
November 2 – 3
China World Hotel
Beijing, China

the adventures of alice & bob

Home | About | Registration & Accommodation | Agenda & Sessions | Sponsors | Contact Us

Wednesday,November 2 | Thursday,November 3

General Sessions | Applications and Development | Cryptography and Architecture | Hackers and Threats | Mobile and Network Security | Trusted and Cloud Computing

Mark Kennedy
Symantec
Topic: Anti-Malware Industry... Cooperating. Are You Serious?

Samir Saklikar
Dennis Moreau
RSA, The Security Division of EMC
Topic: Big Data Techniques for Faster Critical Incident Response

Marc Bown
Trustwave
Topic: APAC Data Compromise Trends

Paula Januszkiewicz
CQURE
Topic: Password Secrets Revealed! All You Want to Know but Are Afraid to Ask

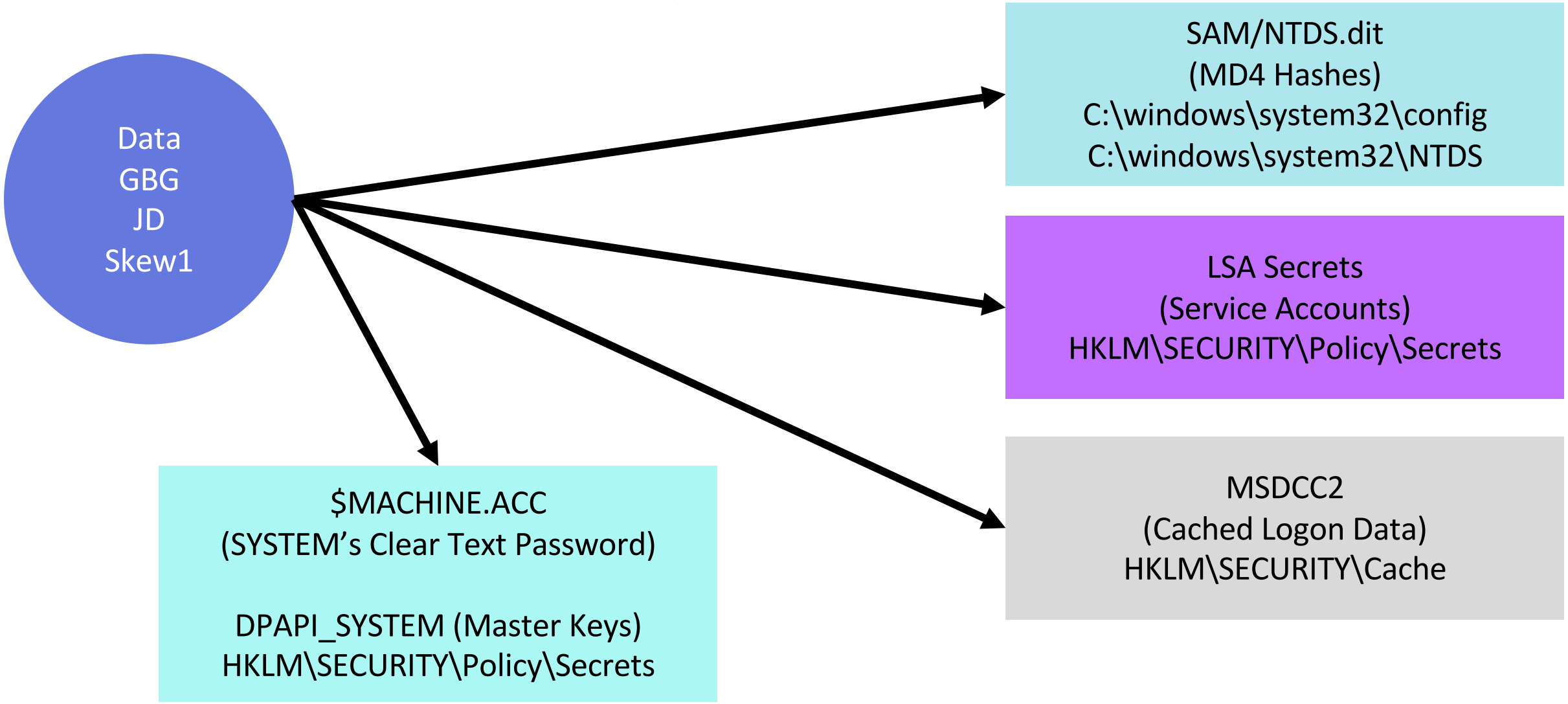

SAMSUNG
Solid State Drive

Definition of credentials

Set of data
that allows other party
to believe me
when I tell who I am

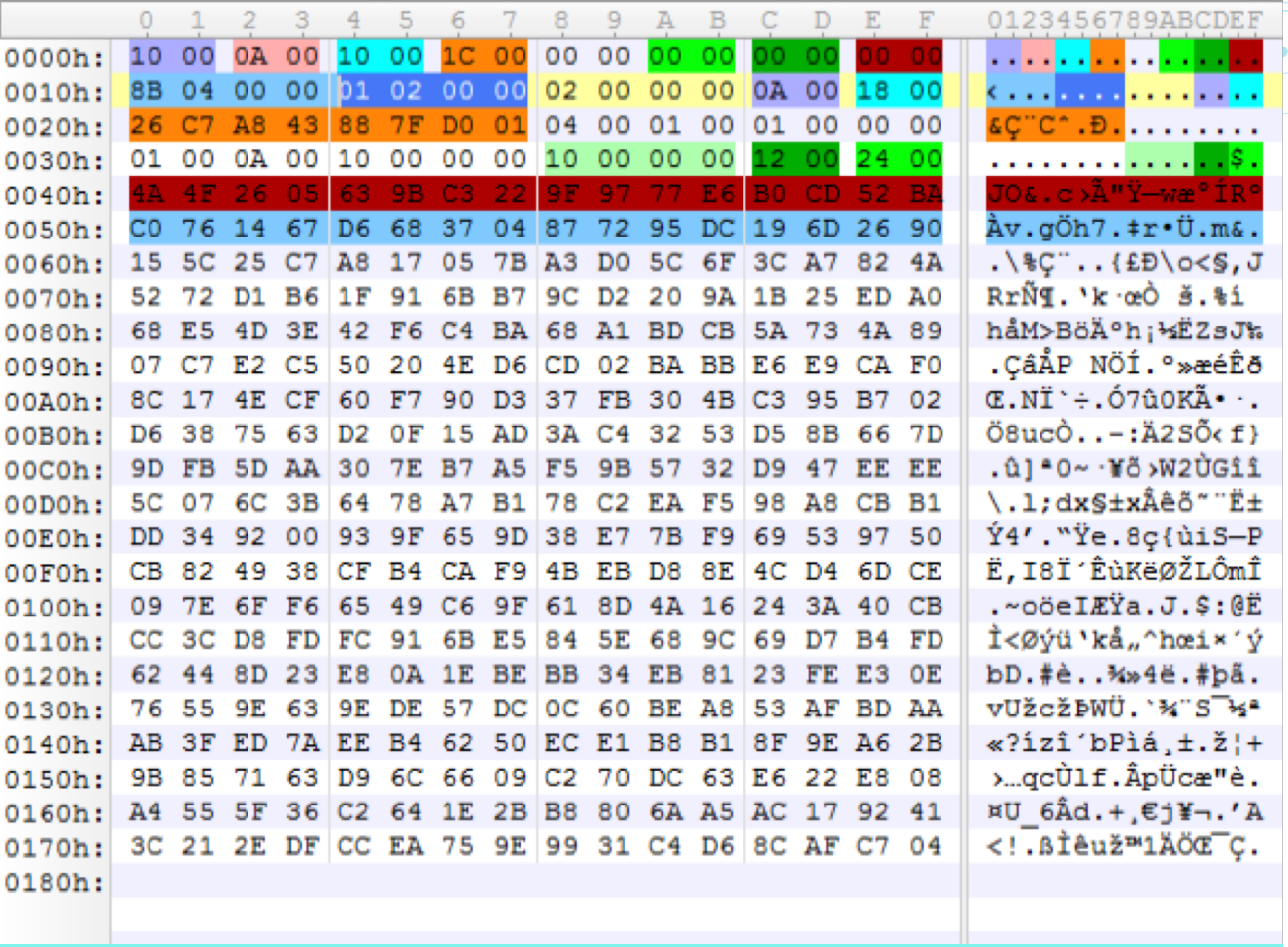
Bootkey:

Class names for keys from HKLM\SYSTEM\CCS\Control\Lsa



More information: http://cqureacademy.com/blog

Are ‘cached credentials’ safe?



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0000h:	10	00	0A	00	10	00	1C	00	00	00	00	00	00	00	00	00	.....
0010h:	8B	04	00	00	01	02	00	00	02	00	00	00	0A	00	18	00	‹.....
0020h:	26	C7	A8	43	88	7F	D0	01	04	00	01	00	01	00	00	00	&Ç¨C^.Ð.
0030h:	01	00	0A	00	10	00	00	00	10	00	00	00	12	00	24	00	.....$.
0040h:	4A	4F	26	05	63	9B	C3	22	9F	97	77	E6	B0	CD	52	BA	JO&.c›Ã"Ÿ—wæ°ÍR°
0050h:	C0	76	14	67	D6	68	37	04	87	72	95	DC	19	6D	26	90	Àv.gÖh7.‡r•Ü.m&.
0060h:	15	5C	25	C7	A8	17	05	7B	A3	D0	5C	6F	3C	A7	82	4A	.\%Ç¨..{£Ð\o<§,J
0070h:	52	72	D1	B6	1F	91	6B	B7	9C	D2	20	9A	1B	25	ED	A0	RrÑ¶.'k·œÒ š.%í
0080h:	68	E5	4D	3E	42	F6	C4	BA	68	A1	BD	CB	5A	73	4A	89	håM>BöÄ°h¡½ËZsJ‰
0090h:	07	C7	E2	C5	50	20	4E	D6	CD	02	BA	BB	E6	E9	CA	F0	.ÇâÅP NÖÍ.°»æéÊð
00A0h:	8C	17	4E	CF	60	F7	90	D3	37	FB	30	4B	C3	95	B7	02	Œ.NÏ`÷.Ó7û0KÃ•·.
00B0h:	D6	38	75	63	D2	0F	15	AD	3A	C4	32	53	D5	8B	66	7D	Ö8ucÒ..-:Ä2SÕ‹f}
00C0h:	9D	FB	5D	AA	30	7E	B7	A5	F5	9B	57	32	D9	47	EE	EE	.û]ª0~·¥õ›W2ÙGîî
00D0h:	5C	07	6C	3B	64	78	A7	B1	78	C2	EA	F5	98	A8	CB	B1	\.l;dx§±xÂêõ˜¨Ë±
00E0h:	DD	34	92	00	93	9F	65	9D	38	E7	7B	F9	69	53	97	50	Ý4'."Ÿe.8ç{ùiS—P
00F0h:	CB	82	49	38	CF	B4	CA	F9	4B	EB	D8	8E	4C	D4	6D	CE	Ë‚I8Ï´ÊùKëØŽLÔmÎ
0100h:	09	7E	6F	F6	65	49	C6	9F	61	8D	4A	16	24	3A	40	CB	.~oöeIÆŸa.J.$:@Ë
0110h:	CC	3C	D8	FD	FC	91	6B	E5	84	5E	68	9C	69	D7	B4	FD	Ì<Øýü'kå„^hœi×´ý
0120h:	62	44	8D	23	E8	0A	1E	BE	BB	34	EB	81	23	FE	E3	0E	bD.#è..¾»4ë.#þã.
0130h:	76	55	9E	63	9E	DE	57	DC	0C	60	BE	A8	53	AF	BD	AA	vUžcžÞWÜ.`¾¨S¯½ª
0140h:	AB	3F	ED	7A	EE	B4	62	50	EC	E1	B8	B1	8F	9E	A6	2B	«?ízî´bPìá¸±.ž¦+
0150h:	9B	85	71	63	D9	6C	66	09	C2	70	DC	63	E6	22	E8	08	›…qcÙlf.ÂpÜcæ"è.
0160h:	A4	55	5F	36	C2	64	1E	2B	B8	80	6A	A5	AC	17	92	41	¤U_6Âd.+¸€j¥¬.'A
0170h:	3C	21	2E	DF	CC	EA	75	9E	99	31	C4	D6	8C	AF	C7	04	<!.ßÌêuž™1ÄÖŒ¯Ç.
0180h:																	

Encrypted Cached Credentials:
Legend

Name	Value	Start	Size	Color	Comment
struct Header h		0h	96	Fg: Bg:	
ushort uname_len	16	0h	2	Fg: Bg:	
ushort domain_len	10	2h	2	Fg: Bg:	
ushort mail_nick_len	16	4h	2	Fg: Bg:	
ushort cn_len	28	6h	2	Fg: Bg:	
ushort u1	0	8h	2	Fg: Bg:	
ushort logon_script_len	0	Ah	2	Fg: Bg:	
ushort profile_path_len	0	Ch	2	Fg: Bg:	
ushort home_dir_len	0	Eh	2	Fg: Bg:	
uint user_sid	1163	10h	4	Fg: Bg:	
uint primary_group_id	513	14h	4	Fg: Bg:	
uint u2	2	18h	4	Fg: Bg:	
ushort group_sids_len	10	1Ch	2	Fg: Bg:	
ushort domain_netbios_name...	24	1Eh	2	Fg: Bg:	
FILETIME last_local_logon	04/25/2015 18:47:22	20h	8	Fg: Bg:	
ushort u3	4	28h	2	Fg: Bg:	
ushort u4	1	2Ah	2	Fg: Bg:	
uint u5	1	2Ch	4	Fg: Bg:	
ushort u6	1	30h	2	Fg: Bg:	
ushort u7	10	32h	2	Fg: Bg:	
uint u8	16	34h	4	Fg: Bg:	
uint u9	16	38h	4	Fg: Bg:	
ushort domain_name_len	18	3Ch	2	Fg: Bg:	
ushort email_len	36	3Eh	2	Fg: Bg:	
byte iv[16]	JO&|c›Ã"Ÿ—wæ°ÍR°	40h	16	Fg: Bg:	
byte cksum[16]	Àv¶gÖh7┘ǂr•Ü ├m&►	50h	16	Fg: Bg:	

Encrypted Cached Credentials
DK = PBKDF2(PRF, Password, Salt, c, dkLen)

Microsoft's implementation: MSDCC2=
PBKDF2(HMAC-SHA1, DCC1, username, 10240, 16)

Cached Logons: It used to be like this…

Windows 2003 / XP

The encryption algorithm is RC4.
The hash is used to verify authentication is calculated as follows:

```
DCC1 = MD4(MD4(Unicode(password)) .
LowerUnicode(username))
is
DCC1 = MD4(hashNTLM . LowerUnicode(username))
```

Usage in the attack

Before the attacks facilitated by pass-the-hash, we can only rejoice the "salting" by the username.

There are a number pre-computed tables for users as Administrator facilitating attacks on these hashes.

Cached Logons: Now it is like this!

Windows Vista / 2008 +

The encryption algorithm is AES128.

The hash is used to verify authentication is calculated as follows:

```
MSDCC2 = PBKDF2(HMAC-SHA1, Iterations,
DCC1, LowerUnicode(username))
```

with DCC 1 calculated in the same way as for 2003 / XP.

Usage in the attack

There is actually not much of a difference with XP / 2003!
No additional salting.

PBKDF2 introduced a new variable: the number of iterations SHA1 with the same salt as before (username).

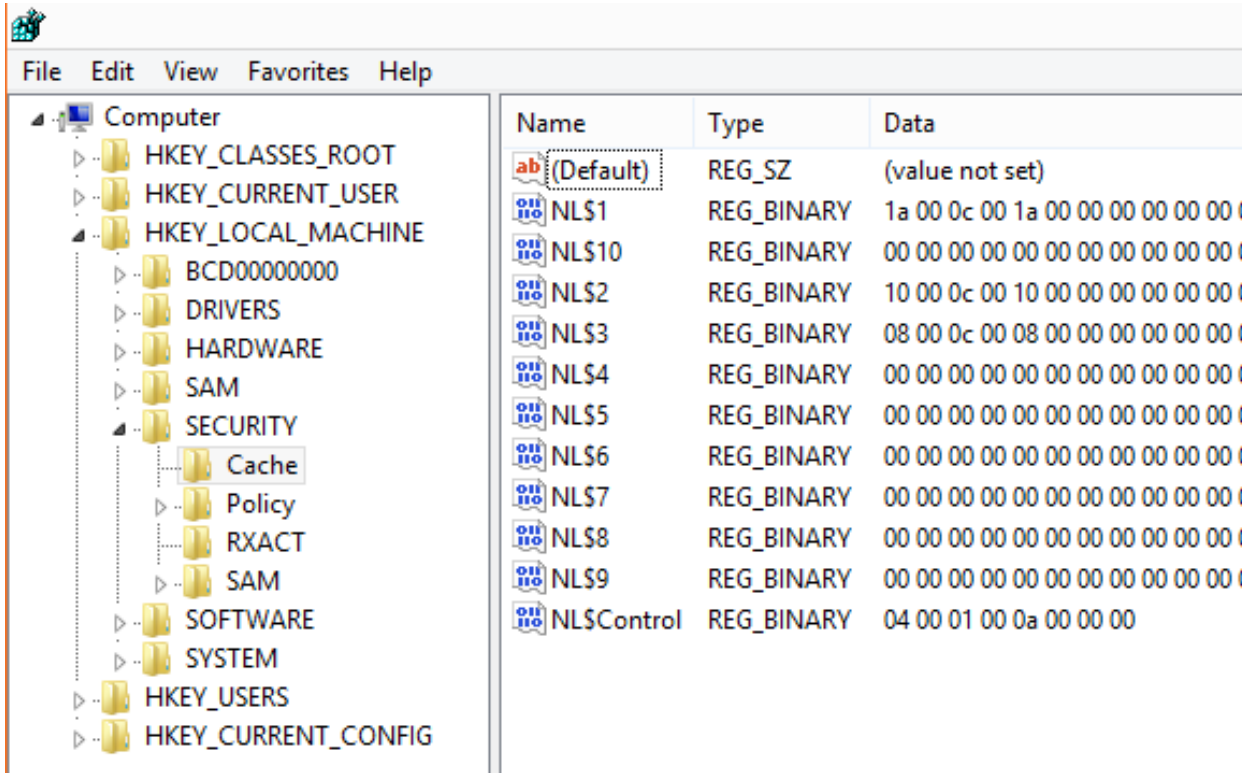
Cached Logons: Iterations

The number of iterations in PBKDF2, it is configurable through the registry:

```
HKEY_LOCAL_MACHINE\SECURITY\Cache
DWORD (32) NL$IterationCount
```

If the number is less than 10240, it is a multiplier by 1024 (20 therefore gives 20480 iterations)

If the number is greater than 10240, it is the number of iterations (rounded to 1024)



Demo: Cached Credentials

+ getting access to user's secrets

Classic Data Protection API

Based on the following components:

Password, data blob, entropy

Is not prone to password resets!

Protects from outsiders when being in offline access

Effectively protects users data

Stores the password history

You need to be able to get access to some of your passwords from the past

Conclusion: OS greatly helps us to protect secrets

Demo: Classic DPAPI

+ getting access to user's secrets in the domain

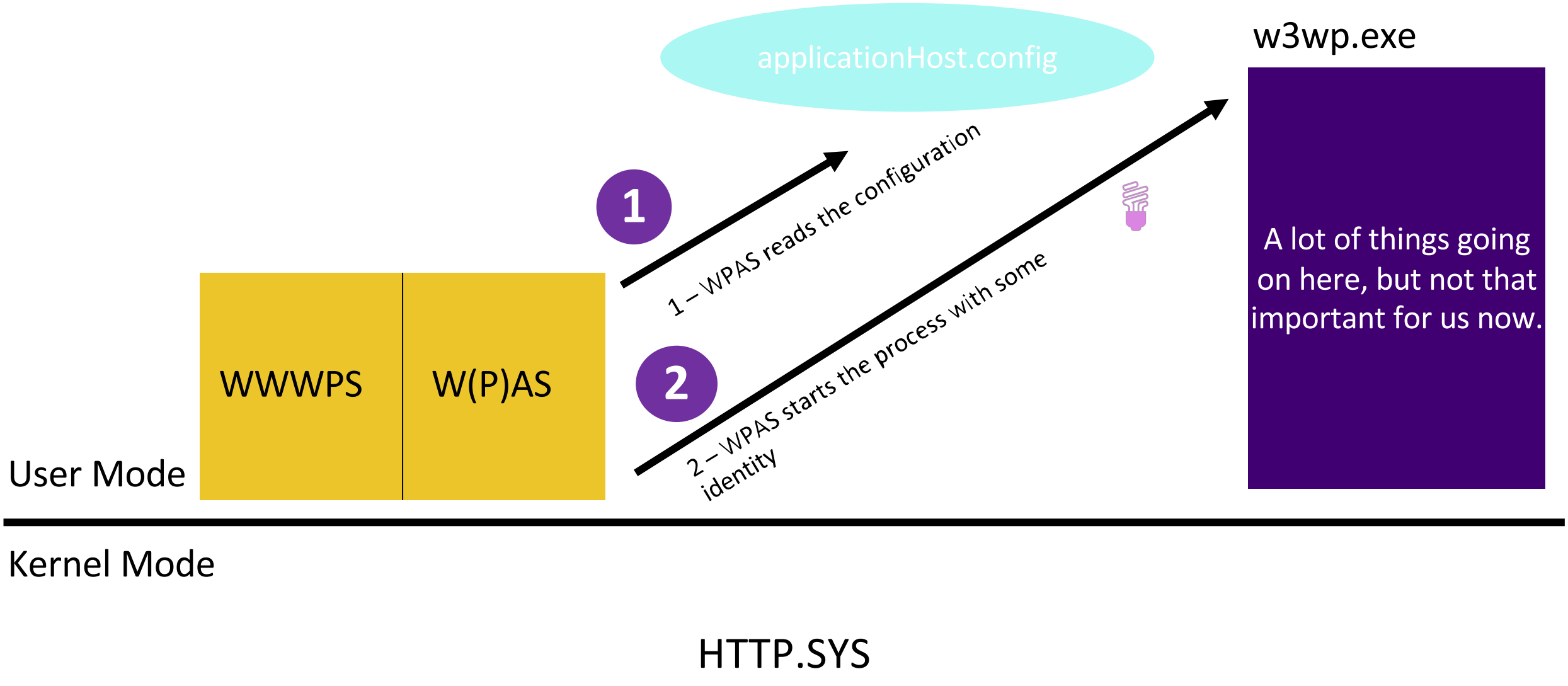
Demo: DPAPI Taken Further

+ Keepass

Demo: RDG Passwords

When centralization should be done with a bit more awareness

IIS Structure



Application Pools

- Used to group one or more Web Applications

Purpose: Assign resources, serve as a security sandbox

- Use Worker Processes (w3wp.exe)

Their identity is defined in Application Pool settings

Process requests to the applications

- Passwords for AppPool identity can be ’decrypted’ even offline

They are stored in the encrypted form in applicationHost.config

Conclusion: IIS relies it’s security on Machine Keys (Local System)

Demo: Application Pools

Getting password from IIS configuration

IISWasKey

+ extracting the data from the registry

Services

Store configuration in the registry

Always need some identity to run the executable!

Local Security Authority (LSA) Secrets

Must be stored locally, especially when domain credentials are used

Can be accessed when we impersonate to Local System

Their accounts should be monitored

If you cannot use gMSA, MSA, use subscription for svc_ accounts (naming convention)

Conclusion: Think twice before using an Administrative account, use gMSA

Demo: Services

Getting password from LSA Secrets

Chasing the obvious: NTDS.DIT, SAM

To perform an analysis on NTDS.DIT the following information sources are needed from the domain controller:

- NTDS.DIT
- Registry hives (at least the SYSTEM hive)

SAM, ntds.dit are stored locally on the server's drive

They **do not** contain Passwords

They use **MD4** as a way of storing them

They are encrypted

The above means: **To read the clear text password you need to struggle!**

Demo: SAM/NTDS.dit

Hash spree - offline

Latest release

2.1.0-20160928

3ea0f0d

2.1 20160922 CQURE Edition for Paula

gentilkiwi released this 8 hours ago

Downloads

mimikatz_trunk.7z	512 KB
mimikatz_trunk.zip	676 KB
Source code (zip)	
Source code (tar.gz)	

Two AMAZING discoveries!

- Kerberos Pre-Authentication

Smart card logon is possible without a smart card

- DPAPI-NG: SID Protected PFX Files

Private keys can be extracted from the PFX files without having a password

Kerberos Pre-Auth

Securing Yourself for a Rainy Day

DPAPI-NG

SID-Protected PFX Files… Unprotected

Credentials Security Takeways

Offline access

Cryptography that relies on keys stored in the registr
is as safe as your offline access.

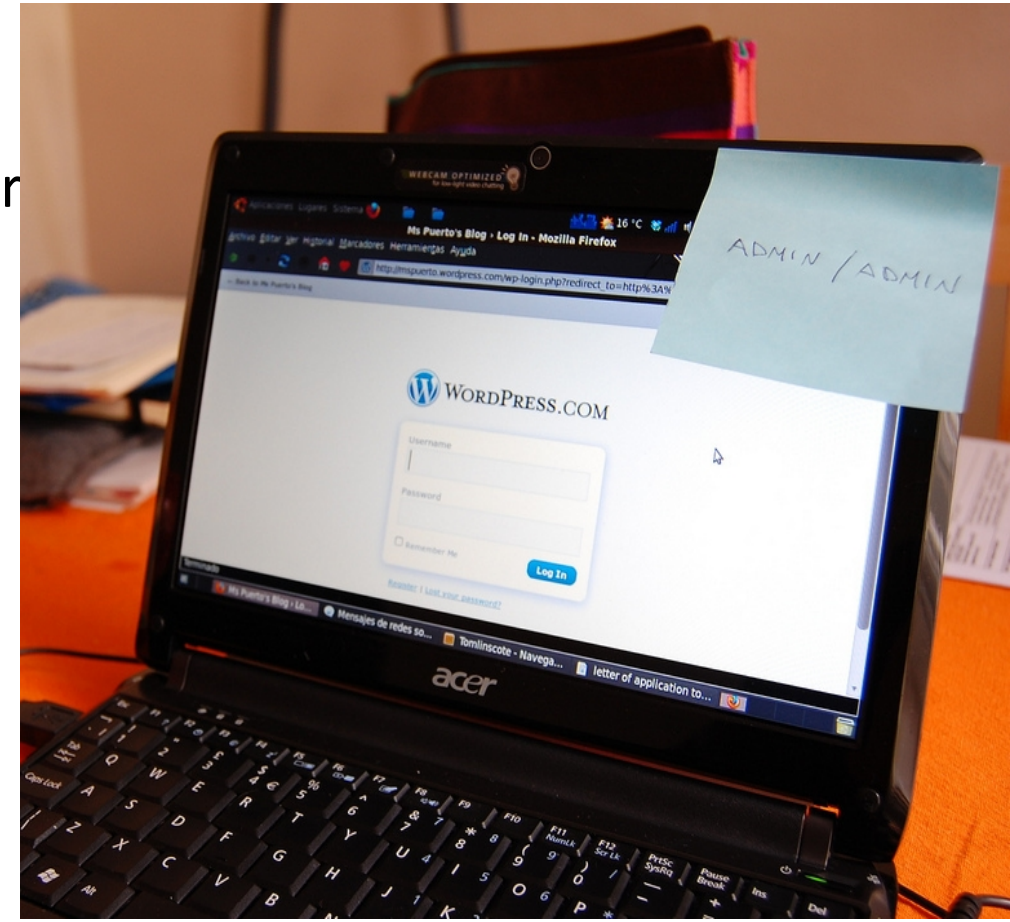
Domain Admins

We all know that they should log on to the Domain Controllers *only*.

Who are they? Can we *trust* them?

Mechanisms are safe

…when extracted. In practice they are as safe as your approach.



RSAConference2019

Thank you!

RSAConference2019

San Francisco | March 4–8 | Moscone Center

BETTER.

SESSION ID:

Understand Credential Security: Important Things You Need to Know About Storing Your Identity

Paula Januszkiewicz

CQURE: CEO, Penetration Tester / Security Expert

CQURE Academy: Trainer

MVP: Enterprise Security, MCT

Microsoft Regional Director

Contact: paula@cqure.us | http://cqure.us