Senior Instructor and Co-Author:
FOR500: Windows Forensics
FOR508: Advanced Forensics and Incident Response

E-mail: chad.tilbury@crowdstrike.com
LinkedIn: Chad Tilbury
Twitter: @chadtilbury

15+ YEARS
Computer Crime Investigations
CrowdStrike • Mandiant • US Air Force
OSI Special Agent

SANS INSTITUTE

CONNECT

CHAD TILBURY
TECHNICAL ADVISOR
CROWDSTRIKE SERVICES
• Priority #1 post-exploitation
  • Domain admin is ultimate goal
• Nearly everything in Windows is tied to an account
  • Difficult to move without one
• Easy and relatively stealthy means to traverse the network
  • Account limitations are rare
• “Sleeper” credentials can provide access after remediation
Evolution of Credential Attack Mitigation

- User Access Control (UAC)
- Managed Service Accounts
- KB2871997

- SSP plaintext password mitigations
- Local admin remote logon restrictions
- Protected Processes
- Restricted Admin
- Domain Protected Users Security Group
- LSA Cache cleanup
- Group Managed Service Accounts

- Credential Guard
- Remote Credential Guard
- Device Guard (prevent execution of untrusted code)
Compromising Credentials: Hashes

The password for each user account in Windows is stored in multiple formats: LM and NT hashes are most well known. TsPkg, WDigest, and LiveSSP can be decrypted to provide plaintext passwords (prior to Win8.1).

How are they acquired and used? Hashes are available in the LSASS process and can be extracted with admin privileges. Once dumped, hashes can be cracked or used immediately in a Pass the Hash attack.

Common tools: Mimikatz • fgdump • gsecdump • Metasploit • SMBshell • PWDumpX • creddump • WCE
## Credential Availability

<table>
<thead>
<tr>
<th>Admin Action</th>
<th>Logon Type</th>
<th>Credentials on Target?</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Console logon</td>
<td>2</td>
<td>Yes*</td>
<td>*Except when Credential Guard is enabled</td>
</tr>
<tr>
<td>Runas</td>
<td>2</td>
<td>Yes*</td>
<td>*Except when Credential Guard is enabled</td>
</tr>
<tr>
<td>Remote Desktop</td>
<td>10</td>
<td>Yes*</td>
<td>*Except for enabled Remote Credential Guard</td>
</tr>
<tr>
<td>Net Use</td>
<td>3</td>
<td>No</td>
<td>Including /u: parameter</td>
</tr>
<tr>
<td>PowerShell Remoting</td>
<td>3</td>
<td>No</td>
<td>Invoke-Command; Enter-PSSession</td>
</tr>
<tr>
<td>PsExec alternate creds</td>
<td>3 + 2</td>
<td>Yes</td>
<td>-u &lt;username&gt; -p &lt;password)</td>
</tr>
<tr>
<td>PsExec w/o explicit creds</td>
<td>3</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Remote Scheduled Task</td>
<td>4</td>
<td>Yes</td>
<td>Password saved as LSA Secret</td>
</tr>
<tr>
<td>Run as a Service</td>
<td>5</td>
<td>Yes</td>
<td>(w/ user account) -- Password saved as LSA Secret</td>
</tr>
<tr>
<td>Remote Registry</td>
<td>3</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

Hash Dumping (Gsecdump)

```
C:\Temp>PsExec
PsExec v2.11
Sysinternals
Microsoft Windows
Copyright (c) 1981-2005
C:\Windows\system32

C:Temp>PsExec
gsecdump.exe -a > 1.txt
```

```
Administrator (current): 500: aad3b435b51404eeaad3b435b51404ee: 7d58f0d8560861d105513c74be
Guest (current-disabled): 501: aad3b435b51404eeaad3b435b51404ee: 31d6cfe0d16ae931b73c59f79
kerbtgt (current-disabled): 502: aad3b435b51404eeaad3b435b51404ee: 70bc2e6bd837b3a18f8c8cb9f
svc-printandscan (current): 1104: aad3b435b51404eeaad3b435b51404ee: 8846f7eae88fbf117ad06bd3
nfury (current): 1105: aad3b435b51404eeaad3b435b51404ee: bfd72559b9a3c32b3c44f6b
mhll (current): 1106: aad3b435b51404eeaad3b435b51404ee: 8846f7eae88fbf117ad06bd3
ctadmin (current): 1107: aad3b435b51404eeaad3b435b51404ee: a5fc9addf4a352074bca862b2e0bac
pcoulson (current): 1108: aad3b435b51404eeaad3b435b51404ee: 8846f7eae88fbf117ad06bd3
nromanoff (current): 1109: aad3b435b51404eeaad3b435b51404ee: 5073bee0c7c3324ab7d4bed5360
svc-bitlocker (current): 1113: aad3b435b51404eeaad3b435b51404ee: 8846f7eae88fbf117ad06bd3
rsydow (current): 1114: aad3b435b51404eeaad3b435b51404ee: ea600320eece92f47f2d29cf37f57a9e
sharepoint (current): 1503: aad3b435b51404eeaad3b435b51404ee: 5bbc1af61e41eb3f827a93293
macsteel (current): 1508: aad3b435b51404eeaad3b435b51404ee: 3d05f1f65da97e2d5632743a1b46348
nicsteel (current): 1509: aad3b435b51404eeaad3b435b51404ee: cce70d80e76e9a826fb97483a0f
thelyons (current): 1510: aad3b435b51404eeaad3b435b51404ee: 266e2bb22867a0a31a8343f327f6e60
davdawson (current): 1511: aad3b435b51404eeaad3b435b51404ee: 7264f7996434c1003921a82730cb8
dawnturner (current): 1512: aad3b435b51404eeaad3b435b51404ee: 93997c01d0d559ed98274b65b2
glebennett (current): 1513: aad3b435b51404eeaad3b435b51404ee: 299551169a5dd31a6db3e5f28f5f9
```
Pass the Hash (Mimikatz)
• Prevent admin account compromise
• Stop remote interactive sessions with highly privileged accounts
• Proper termination of RDP sessions
  • Win8.1+ → force the use of Restricted Admin?
  • Win10 → deploy Remote Credential Guard
• Upgrade to Windows 10
  • Credential Guard
  • TsPkg, WDigest, etc. -- SSO creds obsolescence
  • Domain Protected Users Group (PtH mitigation)
Delegate tokens are powerful authentication resources used for SSO. They allow attackers to impersonate a user’s security context, including over the network.

**How are they acquired and used?** The SeImpersonate privilege lets tokens be copied from processes. The new token can then be used to authenticate as the new user. A target user or service must be logged on or have running processes.

**Common tools:** Incognito • Metasploit • PowerShell • Mimikatz
Token Stealing (Mimikatz)

```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # token::whoami
* Process Token : 3972840  SHIELDBASE\tdungan S-1-5-21-2036804247-3058
324640-2116585241-1107 (15g,25p) Primary
* Thread Token : no token

mimikatz # token::elevate /domainadmin
Token Id : 8
User name : S
SID name : SHIELDBASE\Domain Admins

7892 3157500  SHIELDBASE\rsydow S-1-5-21-2036804247-3058324640-2
116585241-1114 (17g,25p) Primary
-> Impersonated!
* Process Token : 3972840  SHIELDBASE\tdungan S-1-5-21-2036804247-3058
324640-2116585241-1107 (15g,25p) Primary
* Thread Token : 3976490  SHIELDBASE\rsydow S-1-5-21-2036804247-3058
324640-2116585241-1114 (17g,25p) Impersonation <Delegation>
```
Defending Credentials: Tokens

- Prevent admin account compromise
- Stop remote interactive sessions with highly privileged accounts
- Proper termination of RDP sessions
  - Win8.1+ → force the use of Restricted Admin Mode?
  - Win10 → deploy Remote Credential Guard
- Account designation of “Account is Sensitive and Cannot be Delegated” in Active Directory
- Domain Protected Users security group accounts do not create delegate tokens
Compromising Credentials: Cached Credentials

Stored domain credentials to allow logons when domain controller access is unavailable. Most systems cache the last 10 logon hashes by default.

How are they acquired and used? Cached credentials must be cracked. Hashes are salted and case-sensitive, making decryption very slow. These hashes cannot be used for Pass the Hash attacks.

Common tools: cachedump • Metasploit • PWDumpX • creddump
The creddump utilities can extract hashes, cached credentials and LSA Secrets from offline registry hives: github.com/Neohapsis/creddump7
Prevent admin account compromise

Limit number of cached logon accounts
- SOFTWARE\Microsoft\Windows NT\Current Version\Winlogon (cachedlogonscount value)
  - A cachedlogonscount of zero or one is not always the right answer

Enforce password length and complexity rules
- Brute force cracking is required for this attack

Domain Protected Users security group accounts do not cache credentials
Compromising Credentials: LSA Secrets

Credentials stored in the registry to allow services or tasks to be run with user privileges. In addition to service accounts, may also hold application passwords like VPN or auto-logon credentials.

How are they acquired and used? Administrator privileges allow access to encrypted registry data and the keys necessary to decrypt. Passwords are plaintext.

Common tools: Cain • Metasploit • Mimikatz • gsecdump • PWDumpX • creddump • PowerShell
Get-LsaSecret.ps1 from the Nishang PowerShell pentest framework used to dump (and decrypt) LSA Secrets

https://github.com/samratashok/nishang
Defending Credentials: LSA Secrets

- Prevent admin account compromise
- Do not employ services or schedule tasks requiring privileged accounts on low trust systems
- Reduce number of services that require domain accounts to execute
  - Heavily audit any accounts that must be used
- (Group) Managed Service Accounts
Kerberos issues tickets to authenticated users that can be reused without additional authentication. Tickets are cached in memory and are valid for 10 hours.

**How are they acquired and used?** Tickets can be stolen from memory and used to authenticate elsewhere (Pass the Ticket). Further, access to the DC allows tickets to be created for any user with no expiration (Golden Ticket). Service account tickets can be requested and forged, including offline cracking of service account hashes (Kerberoasting).

**Common tools:** Mimikatz • WCE • kerberoast
Pass the Ticket (Mimikatz)

```
kerberos::ptt [0;294e90] 2-0-40e00000-rsydow@krbtgt-SHIELDBASE.LOCAL
```

```
Client: rsydow @ SHIELDBASE.LOCAL
Server: krbtgt/SHIELDBASE.LOCAL @ SHIELDBASE.LOCAL
KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
Ticket Flags 0x40e00000 -> forwardable renewable initial pre_authent
End Time: 2016/11/27 08:32:18 (local)
Renew Time: 2016/12/03 22:32:18 (local)
Session Key Type: AES-256-CTS-HMAC-SHA1-96
```
<table>
<thead>
<tr>
<th>Kerberos Attacks</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pass the Ticket</td>
<td>Steal ticket from memory and pass or import on other systems</td>
</tr>
<tr>
<td>Overpass the Hash</td>
<td>Use NT hash to request a service ticket for the same account</td>
</tr>
<tr>
<td>Kerberoasting</td>
<td>Request service ticket for highly privileged service &amp; crack NT hash</td>
</tr>
<tr>
<td>Golden Ticket</td>
<td>Kerberos TGT for any account with no expiration. Survives full password reset</td>
</tr>
<tr>
<td>Silver Ticket</td>
<td>All-access pass for a single service or computer</td>
</tr>
<tr>
<td>Skeleton Key</td>
<td>Patch LSASS on domain controller to add backdoor password that works for any domain account</td>
</tr>
</tbody>
</table>
Defending Credentials: Tickets

- Credential Guard (Win10+)
  - Domain Protected Users Group (Win8+) – Some attacks
- Remote Credential Guard (Win10+)
  - Restricted Admin (Win8+)
- Long & complex passwords on service accounts (to prevent Kerberoasting)
  - Change service account passwords regularly
  - Group Managed Service Accounts are a great mitigation
- Audit service accounts for unusual activity
- Change KRBTGT password regularly (yearly)
## Kerberos Attack Mitigations

<table>
<thead>
<tr>
<th>Attack Type</th>
<th>Description</th>
<th>Mitigation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pass the Ticket</td>
<td>Steal ticket from memory and pass or import on other systems</td>
<td>Credential Guard; Remote Credential Guard</td>
</tr>
<tr>
<td>Overpass the Hash</td>
<td>Use NT hash to request a service ticket for the same account</td>
<td>Credential Guard; Protected Users Group; Disable RC4 authentication</td>
</tr>
<tr>
<td>Kerberoasting</td>
<td>Request service ticket for highly privileged service &amp; crack NT hash</td>
<td>Long and complex service account passwords; Managed Service Accounts</td>
</tr>
<tr>
<td>Golden Ticket</td>
<td>Kerberos TGT for any account with no expiration. Survives full password reset</td>
<td>Protect domain admin accounts; Change KRBTGT password regularly</td>
</tr>
<tr>
<td>Silver Ticket</td>
<td>All-access pass for a single service or computer</td>
<td>Regular computer account password updates</td>
</tr>
<tr>
<td>Skeleton Key</td>
<td>Patch LSASS on domain controller to add backdoor password to any account</td>
<td>Protect domain admin accounts; Smart card usage for privileged accounts</td>
</tr>
</tbody>
</table>
Active Directory Domain Services (AD DS) database holds all user and computer account hashes (LM/NT) in the domain. Encrypted, but algorithm is well known and easy to defeat.

How is it acquired and used? Located in the Windows\NTDS folder on the domain controller. The file is locked, so admin access is required to load a driver to access raw disk, or use the Volume Shadow Copy Service.

Common tools: ntdsutil • VSSAdmin • NTDSXtract • VSSOwn.vbs • PowerShell • ntdsdump
Stealing NTDS.DIT

CommandProcess: conhost.exe Pid: 141716
CommandHistory: 0x1b8f80 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 12 LastAdded: 11 LastDisplayed: 11
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x60
Cmd #0 @ 0x196970: vssadmin list shadows
Cmd #1 @ 0x1bd240: cd \nCmd #2 @ 0x1b9290: dir
Cmd #3 @ 0x1bd260: cd temp
Cmd #4 @ 0x1b92b0: dir
Cmd #5 @ 0x19c6a0: copy\\\%GLOBALROOT\%Device\HarddiskVolumeShadowCopy49\windows\system32\config\SYSTEM .
Cmd #6 @ 0x19c760: dir
Cmd #7 @ 0x19c780: copy\\\%GLOBALROOT\%Device\HarddiskVolumeShadowCopy49\windows\system32\config\SAM .
Cmd #8 @ 0x19c830: copy\\\%GLOBALROOT\%Device\HarddiskVolumeShadowCopy49\windows\ntds\ntds.dit .
Cmd #9 @ 0x1clab0: dir
Defending Credentials: NTDS.DIT

Don’t allow Domain Admin accounts to be compromised.
Credential Attack Detection
“As any pass-the-ticket attack, the attacker replays the golden ticket in a standard Kerberos protocol. Therefore, there is no clear indication of such attack in Windows logs.”
“Golden Ticket events may have one of these issues:
The Account Domain field is blank when it should be DOMAIN
The Account Domain field is DOMAIN FQDN when it should be DOMAIN.” –Sean Metcalf, adsecurity.org
As an example...

Kerberoasting uses RC4 encryption downgrade

(but almost no one logs these events)
Credential Attack Detection

Authentication Auditing
- Mapping Admin$ Shares
- PsExec
- Scheduled Tasks
- VSSAdmin
- RDP/VPN activity

Tool Artifacts
- New Services
- Random File / Host names
- Code Injection
- Crashes and Security Alerts

Behavioral Analysis
- Local Admin Account Use
- Domain Admin Anomalies
- Service Account Anomalies
- Workstation-to-workstation connections
Event logs are critical for detection
- Authentication events (EID 4624, 4762, 4648, 4720, etc.)
- New services (EID 7045)
- Application and Process Crashes
- Failed and anomalous SMB activity (EID 5140)
- AV / Security logs
- Domain Protected User security group logs
  - Applications and Services Logs\Microsoft\Windows\Microsoft\Authentication
- Process tracking
  - Command line captures
  - PowerShell auditing
Credential Attack Detection: Pass the Hash

Event 4776, Microsoft Windows security auditing.

- **Logon Account:** Administrator
- **Source Workstation:** jn5mFjAjPsFApn3v
- **Error Code:** 0xC000006A

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Date and Time</th>
<th>Event ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit Failure</td>
<td>4/3/2012 12:30:43 PM</td>
<td>4776</td>
</tr>
<tr>
<td>Audit Failure</td>
<td>4/3/2012 12:33:13 PM</td>
<td>4776</td>
</tr>
<tr>
<td>Audit Failure</td>
<td>4/3/2012 12:34:56 PM</td>
<td>4776</td>
</tr>
<tr>
<td>Audit Failure</td>
<td>4/3/2012 12:36:49 PM</td>
<td>4776</td>
</tr>
<tr>
<td>Audit Success</td>
<td>4/3/2012 12:39:34 PM</td>
<td>4776</td>
</tr>
<tr>
<td>Audit Success</td>
<td>4/3/2012 12:39:34 PM</td>
<td>4624</td>
</tr>
</tbody>
</table>

Event 4624, Microsoft Windows security auditing.

- **Account Name:** SRL-Helpdesk
- **Account Domain:** WKS-WIN732BITA
- **Logon GUID:** {00000000-0000-0000-0000-000000000000}
- **Process Information:**
  - **Process ID:** 0x0
  - **Process Name:** -
- **Network Information:**
  - **Workstation Name:** 4y6j74CKqiNXcc1
  - **Source Network Address:** 10.3.58.7
  - **Source Port:** 4508
- **Detailed Authentication Information:**
  - **Logon Process:** Ntlm5sp
  - **Authentication Package:** NTLM

An account was successfully logged on.
Credential Attack Detection: PsExec and fgdump

Initiation of two near-simultaneous services by helpdesk account
Credential Attack Detection: LSASS Crash

** Review and correlate your Anti-Virus logs **

- **Application Event Log**
  - Date: 5/27/2010
  - Source: Winlogon
  - Time: 1:02:06 AM
  - Category: None
  - Type: Error
  - Event ID: 1015
  - User: N/A
  - Computer: SANSFORE-2112
  - Description:
    
    A critical system process, C:\WINDOWS\system32\lsass.exe, failed with status code 00000005. The machine must now be restarted.
    
    For more information, see Help and Support Center at http://go.microsoft.com/fwlink/events.asp.

- **System Event Log**
  - Date: 5/27/2010
  - Source: Microsoft Antimalware
  - Time: 1:01:29 AM
  - Category: None
  - Type: Warning
  - Event ID: 1006
  - User: N/A
  - Computer: SANSFORE-2112
  - Description:
    
    Microsoft Antimalware has detected spyware or other potentially unwanted software.
    For more information please see the following:
    Name: HackTool:Win32/FWDump.A
    ID: 2147617810
    Severity: Medium
    Category: Tool
    Path: file:C:\WINDOWS\LsaExt.dll, file:C:\WINDOWS\LsaExt.dll
Credential Attack Detection: Capturing Command Lines

Event Properties - Event 4688, Microsoft Windows security auditing.

A new process has been created.

Subject:
- Security ID: [Redacted]
- Account Name: [Redacted]
- Account Domain: [Redacted]
- Logon ID: 0x46068

Process Information:
- New Process ID: 0x210
- New Process Name: \powershell.exe
- Token Elevation Type: TokenElevationTypeDefault (1)
- Creator Process ID: 0x4e0

Process Command Line: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" --hidden --enc cGFyYW0gKCRDb2ldXWlckXhbwWugPSAiiLislCRGaWxlUGF0aCA9ICluXEFWcvGxpY2F0aXZl

dmVudG9yey5jci3YIKQOKDQpDpZXQtdZ1ppZjQZWNOIC1xdVWyeSAiU0VRUNUICogR

W4Z4M19Qmc9kdWN0NiAtY29tH2V0YyW1IClCRDb2ldXWlkXhbwWugfACAnCnYvncn
N0FIZlboRvcRBIAROkzcv2sZWN0LV9iamVjdCBQU0lvdXB1BGMvTmFtZSxxWZkvb3I3
ZXIzaW0uLEnhHRpb2s4GVyY3JpcHRpb24sSW5zdGFsb2ErhGUuSW5zdGFsbExvY2F
uc3RhbGxGb37vy2UsUGFja2Fnb2JlU3ZlY3Jvb283JmluZGluZ3J5
WFucGVuZCw=

Token Elevation Type indicates the type of token that was assigned to the new process in accordance with User Account Control policy.
Credential Attack Detection: Other Data Sources

• Registry changes
  • Disabled computer account pwd updates (Silver Tickets)
    ```
    SYSTEM\CurrentControlSet\Services\Netlogon\Parameters
    DisablePasswordChange=1
    ```
  • Enabled WDigest credentials (post Win8.1)
    ```
    SYSTEM\CurrentControlSet\Control\SecurityProviders\Wdigest
    UseLogonCredential=1
    ```

• Memory Analysis
  • Process injection
  • Loaded drivers
• Kernel-level security agent detections
• Behavioral Analytics
Credential Best Practices
Restrict and Protect Privileged Domain Accounts

- Reduce the number of Domain/Enterprise Admins
- Enforce multi-factor authentication (MFA) for all network and cloud admin accounts
- Separate administrative accounts from user accounts for administrative personnel
- Create specific administrative workstation hosts for administrators
- Use the **Domain Protected Users security group**!
Limit Local Admin Accounts

• Don’t give users admin
• Unique and complex passwords for local admin (LAPS)
• Deny network logons for local accounts

Audit account usage and monitor for anomalies
Use a Tiered Administrative Access Model

- Administration of AD
- Servers and Applications
- Workstations and Devices

Image source: Securing Privileged Access Reference Material by Corey Plett (Microsoft)
Best Practices: Reduce the Credential Attack Surface

• Audit and limit the number of services running as system and domain accounts
  • Utilize Group Managed Service Accounts
  • ... or regularly change and use long & complex passwords
• Upgrade to Windows 10 /Server 2016
  • Enable Credential Guard & Remote Credential Guard
• Force LSASS as protected process on legacy Win8.1
• Establish remote connections using network logon instead of interactive logon when possible
Best Practices: Reduce the Credential Attack Surface (2)

- Limit workstation to workstation communication
  - Restrict inbound NetBIOS, SMB traffic using the Windows Firewall
  - ... or VLAN segmentation of workstations
  - So many hack tools leverage SMB authentication
  - Is workstation to workstation RDP really necessary?

- Enable stricter Kerberos security
  - Disable LM & NTLM (force Kerberos)
  - Short validity for tickets
  - No account delegation
## Increase Awareness of New Attacks

Chart by Benjamin Delpy: [https://goo.gl/1K3AC7](https://goo.gl/1K3AC7)

<table>
<thead>
<tr>
<th>Windows XP/2003</th>
<th>LM</th>
<th>NTLM</th>
<th>SHA1</th>
<th>NTLM</th>
<th>SHA1</th>
<th>Root</th>
<th>DPAPI</th>
<th>tspkg</th>
<th>off</th>
<th>on</th>
<th>wdigest</th>
<th>off</th>
<th>on</th>
<th>pass</th>
<th>PIN</th>
<th>tickets</th>
<th>eKeys</th>
<th>livessp</th>
<th>ssp</th>
<th>dpapi</th>
<th>credman</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Windows Vista/2008 &amp; 7/2008r2</th>
<th>LM</th>
<th>NTLM</th>
<th>SHA1</th>
<th>NTLM</th>
<th>SHA1</th>
<th>Root</th>
<th>DPAPI</th>
<th>tspkg</th>
<th>off</th>
<th>on</th>
<th>wdigest</th>
<th>off</th>
<th>on</th>
<th>pass</th>
<th>PIN</th>
<th>tickets</th>
<th>eKeys</th>
<th>livessp</th>
<th>ssp</th>
<th>dpapi</th>
<th>credman</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Windows 8/2012</th>
<th>LM</th>
<th>NTLM</th>
<th>SHA1</th>
<th>NTLM</th>
<th>SHA1</th>
<th>Root</th>
<th>DPAPI</th>
<th>tspkg</th>
<th>off</th>
<th>on</th>
<th>wdigest</th>
<th>off</th>
<th>on</th>
<th>pass</th>
<th>PIN</th>
<th>tickets</th>
<th>eKeys</th>
<th>livessp</th>
<th>ssp</th>
<th>dpapi</th>
<th>credman</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Windows 8.1/2012r2</th>
<th>LM</th>
<th>NTLM</th>
<th>SHA1</th>
<th>NTLM</th>
<th>SHA1</th>
<th>Root</th>
<th>DPAPI</th>
<th>tspkg</th>
<th>off</th>
<th>on</th>
<th>wdigest</th>
<th>off</th>
<th>on</th>
<th>pass</th>
<th>PIN</th>
<th>tickets</th>
<th>eKeys</th>
<th>livessp</th>
<th>ssp</th>
<th>dpapi</th>
<th>credman</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain Account</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain Protected Users</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Windows 8.1 vault for user’s authentication

- PIN: code
- Picture: pass
- Fingerprint: pass

<table>
<thead>
<tr>
<th>PIN</th>
<th>Picture</th>
<th>Fingerprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>code</td>
<td>pass</td>
<td>gestures</td>
</tr>
</tbody>
</table>

#### Notes:
1. can need an unlock on NTS, not available with smartcard
2. tspkg is not installed by default on XP, not available on 2003
3. tspkg is off by default (but needed for SSO with remoteapps/ts), wdigest too
4. PIN code when SmartCard used for native Logon
5. PIN code is NOT encrypted in memory (XP/2003)
6. When accessed/used by owner
7. When local admin, UAC and after unlock

---

[SANS DFIR](https://www.sans.org/dfir)