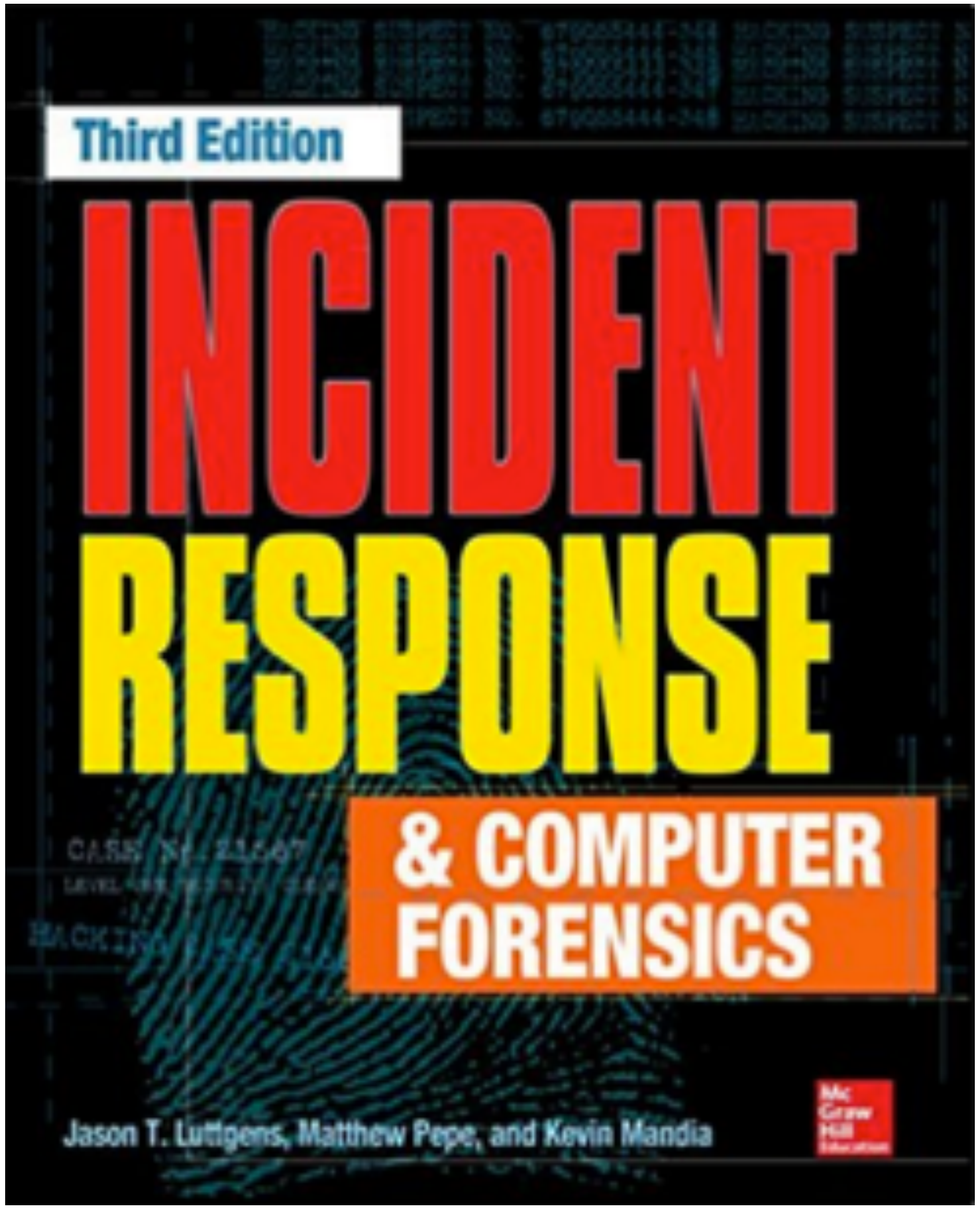
CNIT 121: Computer Forensics



12 Investigating Windows Systems

- NTFS and file system analysis
- Windows prefetch
- Event logs
- Scheduled tasks
- The registry
- Other artifacts of interactive sessions
- Memory forensics
- Alternative persistence mechanisms

NTFS and File System Analysis

NTFS and FAT

- **FAT was the old file system used by MS-DOS, Windows 95, Windows 98**
- **NTFS was the replacement**

Master File Table (MFT)

- **Defines how disk space is allocated and utilized**
- **How files are created and deleted**
- **How metadata is stored and updates**

MFT Contents

- Primary source of metadata in NTFS
- Contains or references everything about a file
  - Timestamps
  - Size
  - Attributes (such as permissions)
  - Parent directory
  - Contents

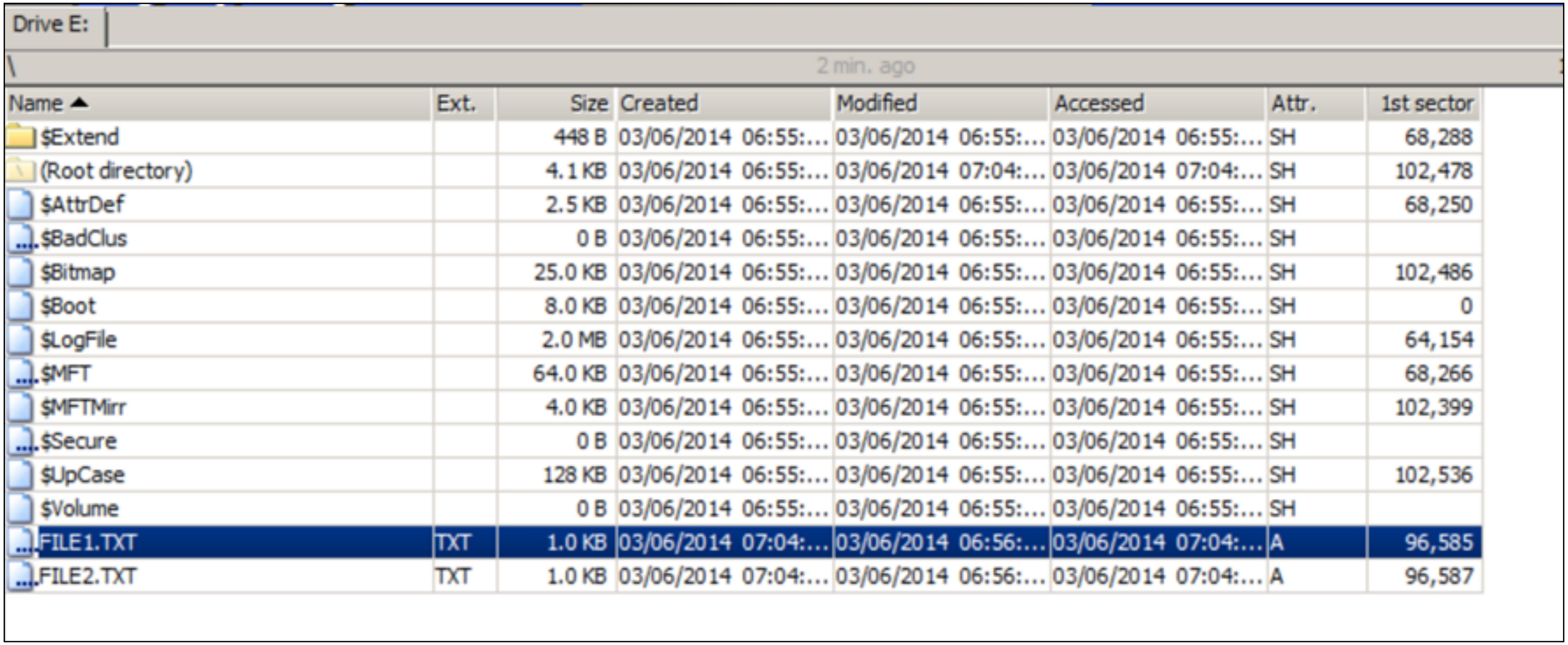
The Evidence

- **Each NTFS volume has its own MFT**
- **Stored in the volume root as a file named $MFT**
- **You need raw disk access to acquire $MFT**
  - **It's not accessible through Windows Explorer or standard API calls**

MFT Structure

- **On a standard hard drive with 512-byte sectors**
- **A series of 1024-byte records or "entries"**
- **One for each file and directory on a volume**
- **First 16 entries are reserved for essential NTFS artifacts**
    - **$MFT itself, $LogFile, and more**

MFT in WinHex



Drive E:

\ 2 min. ago

Name	Ext.	Size	Created	Modified	Accessed	Attr.	1st sector
$Extend		448 B	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	68,288
(Root directory)		4.1 KB	03/06/2014 06:55:...	03/06/2014 07:04:...	03/06/2014 07:04:...	SH	102,478
$AttrDef		2.5 KB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	68,250
$BadClus		0 B	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	
$Bitmap		25.0 KB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	102,486
$Boot		8.0 KB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	0
$LogFile		2.0 MB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	64,154
$MFT		64.0 KB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	68,266
$MFTMirr		4.0 KB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	102,399
$Secure		0 B	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	
$UpCase		128 KB	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	102,536
$Volume		0 B	03/06/2014 06:55:...	03/06/2014 06:55:...	03/06/2014 06:55:...	SH	
FILE1.TXT	TXT	1.0 KB	03/06/2014 07:04:...	03/06/2014 06:56:...	03/06/2014 07:04:...	A	96,585
FILE2.TXT	TXT	1.0 KB	03/06/2014 07:04:...	03/06/2014 06:56:...	03/06/2014 07:04:...	A	96,587

MFT Entry Contents

- **Record type (file or directory)**
- **Record # (integer)**
- **Parent record #**
- **Active/Inactive flag**
  - **Deleted files are inactive**
- **Attributes (metadata)**

Attributes

- **$STANDARD_INFORMATION**
- **$FILE_NAME**
- **$DATA**

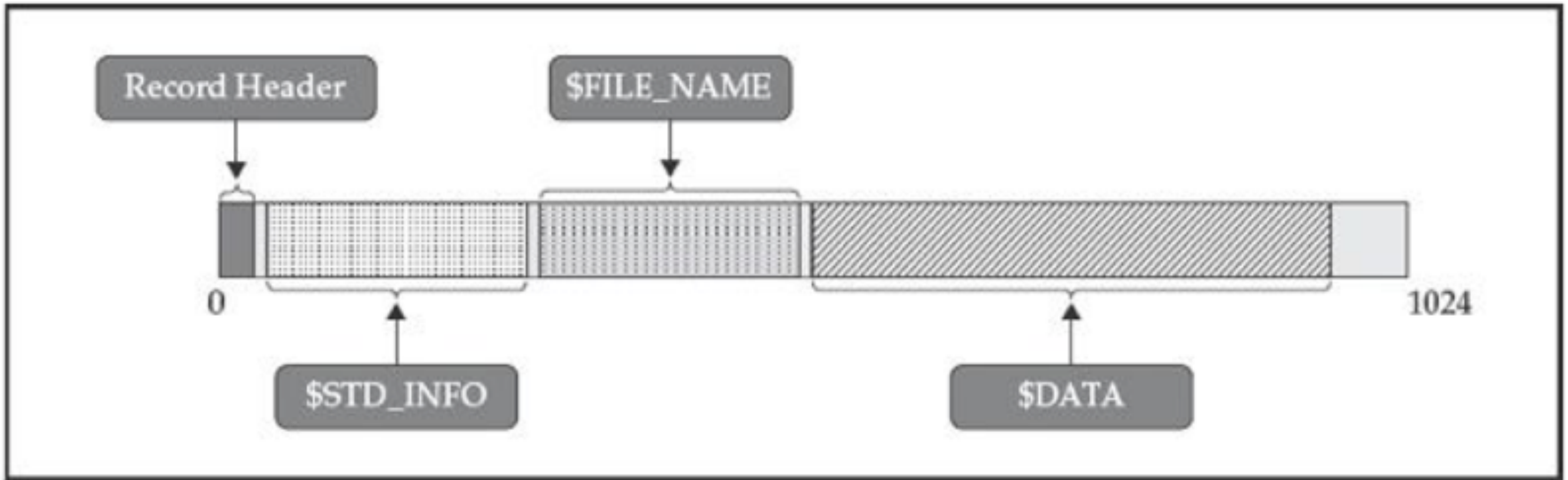


Figure 12-1. Overview of the structure of an MFT file record

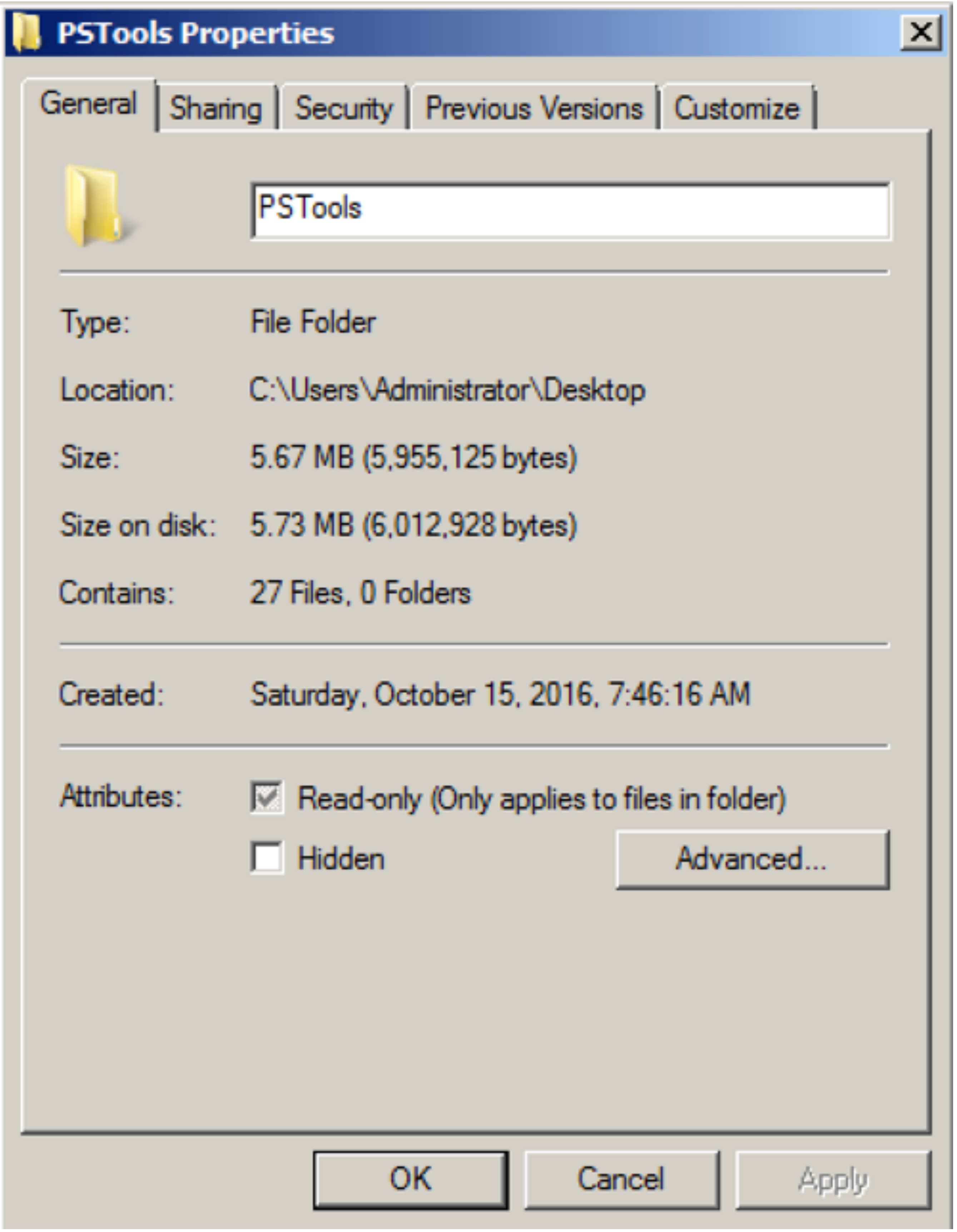
Deleted Files

- Deleting a file causes its MFT record to be marked "inactive"
- Nothing else is changed, until this record is re-used
- The file's contents and its metadata can be recovered
- But NTFS will always re-use an existing MFT entry before creating a new one
- So inactive entries only last for seconds or minutes on the operating system volume

Timestamps

- **MACE timestamps**
  - **Modified, Accessed, Created, Entry Modified**
- **An MFT entry will always have at least two sets of attributes containing MACE timestamps**
  - **STANDARD_INFORMATION (also known as $SIA or $SI)**
  - **FileName (also known as FNA, FILE_NAME, or $FN)**

- These are Standard Information ($SI) timestamps
  - Created
  - Accessed
  - Modified
- Entry Modified timestamp not visible in Windows Explorer
- Forensic tools like SleuthKit, EnCase, and FTK show it



PSTools

Type: File Folder

Location: C:\Users\Administrator\Desktop

Size: 5.67 MB (5,955,125 bytes)

Size on disk: 5.73 MB (6,012,928 bytes)

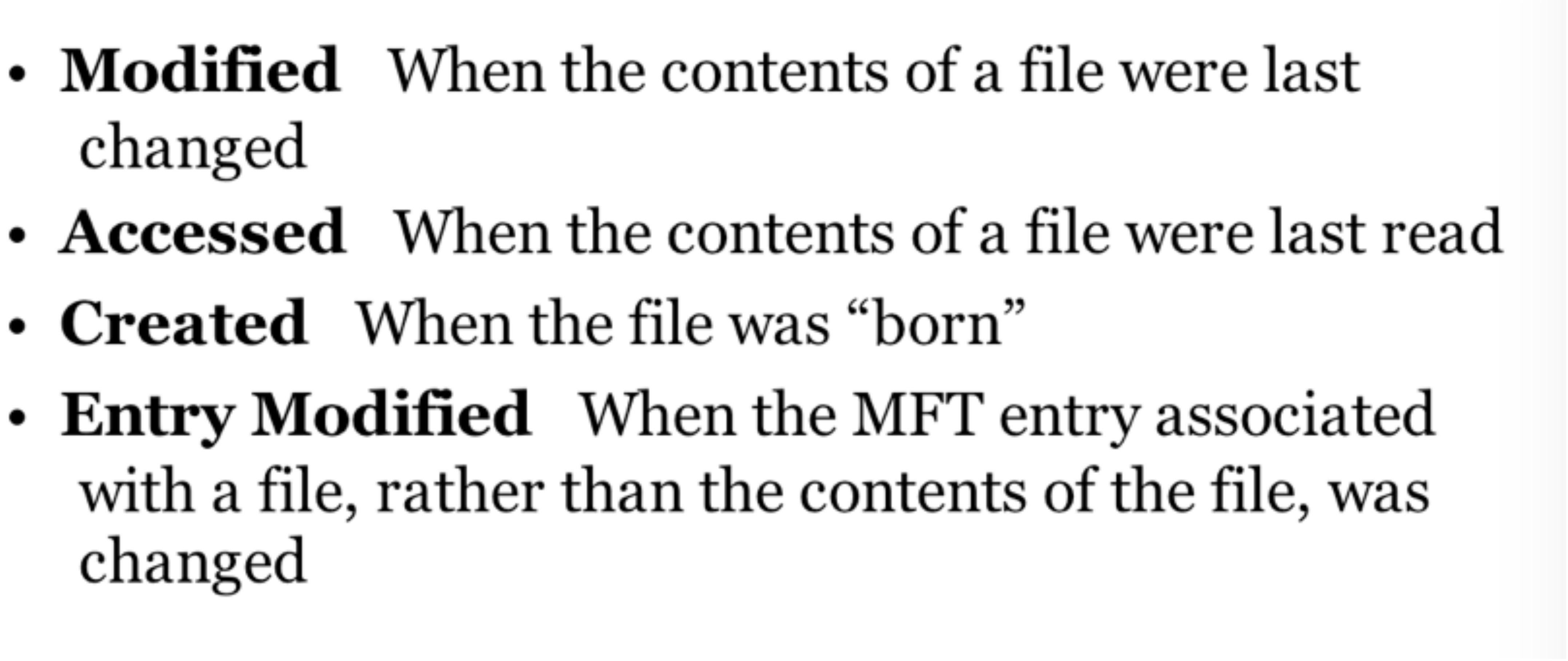
Contains: 27 Files, 0 Folders

Created: Saturday, October 15, 2016, 7:46:16 AM

Attributes: ☑ Read-only (Only applies to files in folder)

□ Hidden

MACE Timestamps



- **Modified** When the contents of a file were last changed
- **Accessed** When the contents of a file were last read
- **Created** When the file was "born"
- **Entry Modified** When the MFT entry associated with a file, rather than the contents of the file, was changed

Accessed Timestamp

- **Versions of Windows after Windows XP no longer update the Accessed timestamp by default**
- **It can be enabled with a registry change, but even when it's enabled, NTFS may delay updates by up to an hour**
  - **Link Ch 12a**

$FN Timestamps

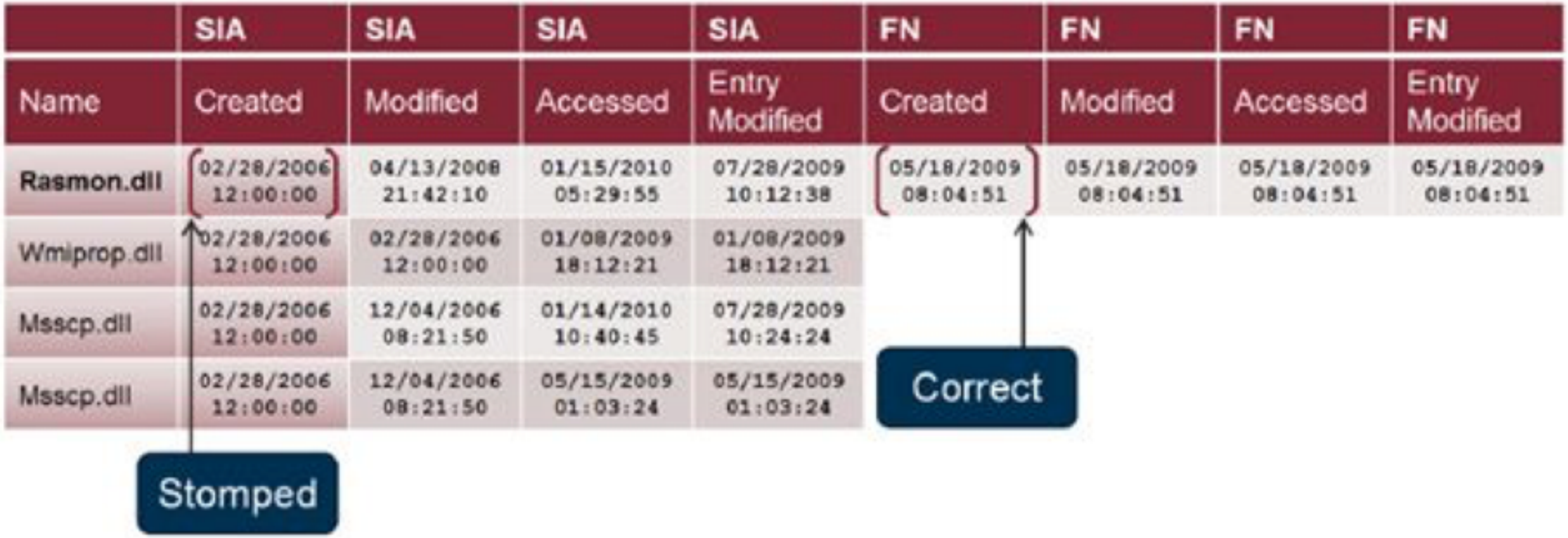
- Refer to the MFT entry for the filename itself
- NTFS actually maintains multiple sets of file name attributes
  - Full, case-sensitive long filename
  - MS-DOS 8.3 short file name

Time-Stomping

- **Only the $SI timestamps are available to user applications through the Windows API**
- **Programs can only alter those timestamps**
  - **A processes called "time-stomping"**
- **Setmace can alter all the timestamps (link Ch 12b)**
- **Malware droppers and installers often automate this process, inserting timestamps from system files to hide in the timeline**

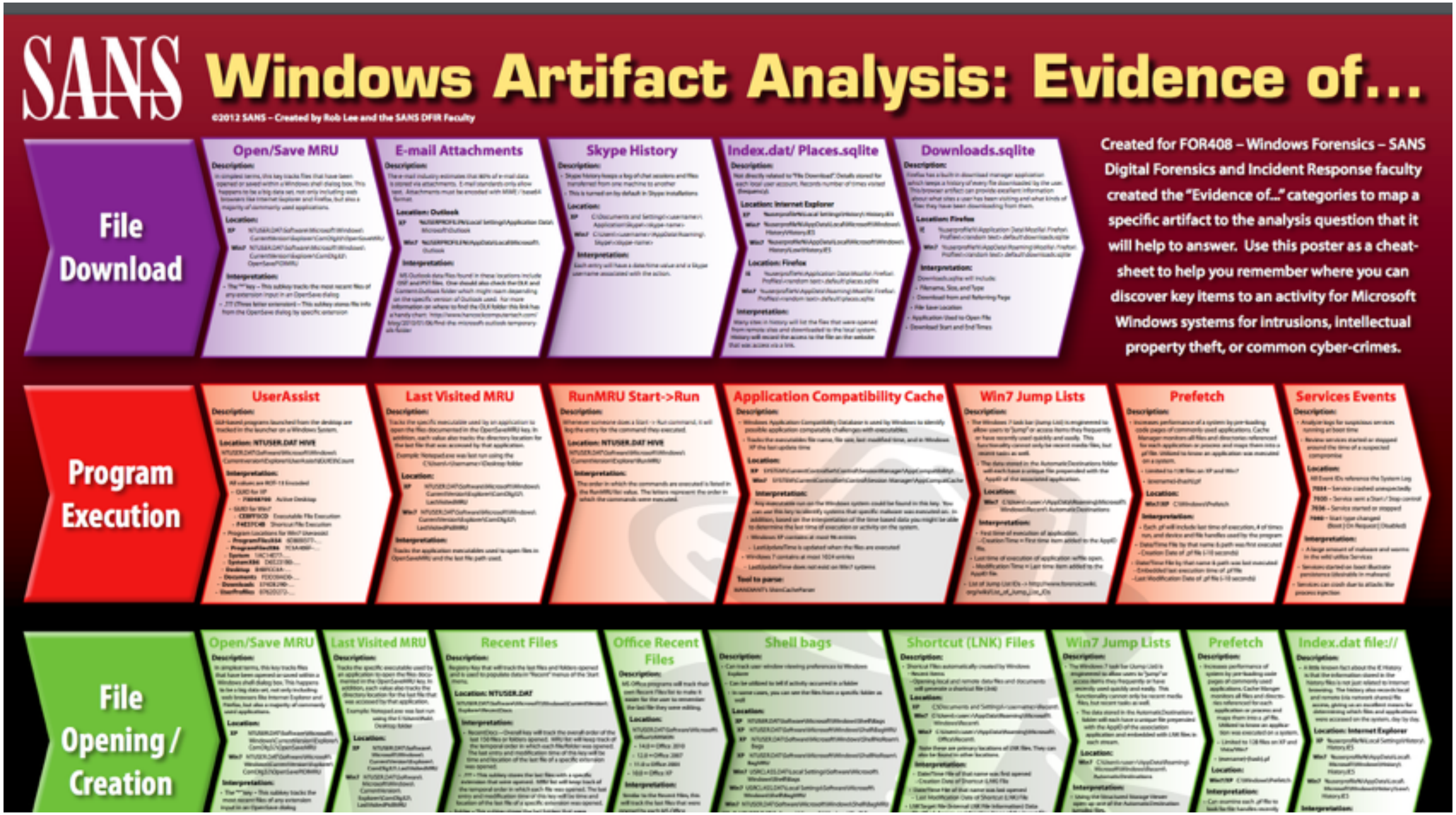
$SI and $FN Timestamps

- **$SI timestamps are easily altered**
- **$FN timestamps require a complex and indirect process to modify**
- **Inconsistencies may remain between the $SI and $FN timestamps**



	SIA	SIA	SIA	SIA	FN	FN	FN	FN
Name	Created	Modified	Accessed	Entry Modified	Created	Modified	Accessed	Entry Modified
Rasmon.dll	02/28/2006 12:00:00	04/13/2008 21:42:10	01/15/2010 05:29:55	07/28/2009 10:12:38	05/18/2009 08:04:51	05/18/2009 08:04:51	05/18/2009 08:04:51	05/18/2009 08:04:51
Wmiprop.dll	02/28/2006 12:00:00	02/28/2006 12:00:00	01/08/2009 18:12:21	01/08/2009 18:12:21				
Msscp.dll	02/28/2006 12:00:00	12/04/2006 08:21:50	01/14/2010 10:40:45	07/28/2009 10:24:24				
Msscp.dll	02/28/2006 12:00:00	12/04/2006 08:21:50	05/15/2009 01:03:24	05/15/2009 01:03:24				

Figure 12-3. Timestamp manipulation example



SANS **Windows Artifact Analysis: Evidence of...**

©2012 SANS – Created by Rob Lee and the SANS DFIR Faculty

File Download

Open/Save MRU

E-mail Attachments

Skype History

Index.dat/ Places.sqlite

Downloads.sqlite

Created for FOR408 – Windows Forensics – SANS Digital Forensics and Incident Response faculty created the "Evidence of..." categories to map a specific artifact to the analysis question that it will help to answer. Use this poster as a cheat-sheet to help you remember where you can discover key items to an activity for Microsoft Windows systems for intrusions, intellectual property theft, or common cyber-crimes.

Program Execution

UserAssist

Last Visited MRU

RunMRU Start->Run

Application Compatibility Cache

Win7 Jump Lists

Prefetch

Services Events

File Opening / Creation

Open/Save MRU

Last Visited MRU

Recent Files

Office Recent Files

Shell bags

Shortcut (LNK) Files

Win7 Jump Lists

Prefetch

Index.dat file://

- **Link Ch 12c**

Data Runs

- **$DATA attribute lists all clusters with the file's contents**

- **May not be contiguous (fragmented file)**

  - **Lists "data runs" that must be assembled together to get the complete file**

Resident Data

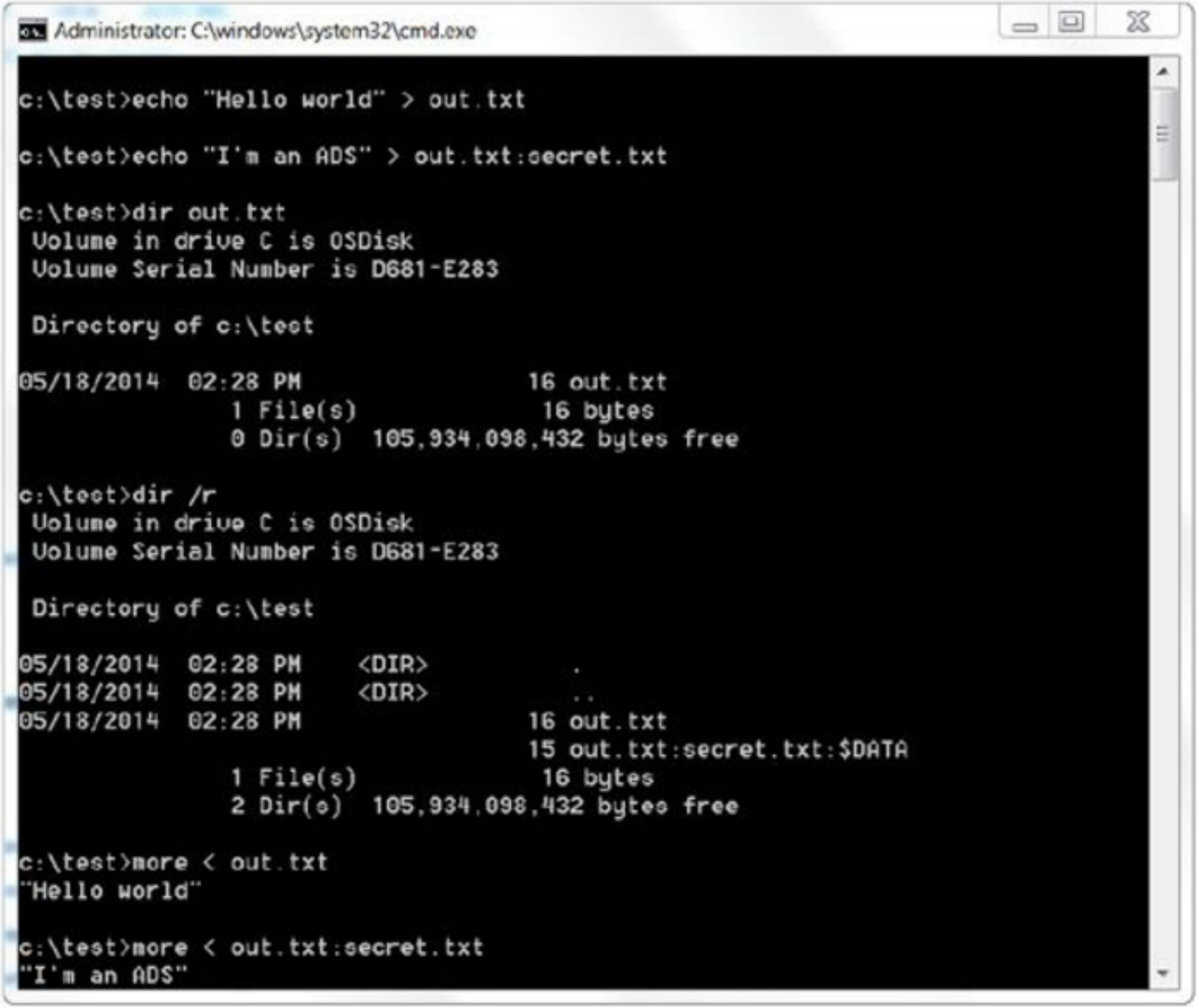
- **MFT entry contains 1024 bytes**
- **That's enough room to store complete data for small files (up to 700 or 800 bytes) in the MFT**
- **These are called "Resident files"**
- **Set the Resident flag in the MFT entry**

MFT Slack Space

- **MFT may contain leftovers from previously resident data**
- **This happens if a file was small enough to be resident and then expanded to be too large to remain resident**

Alternate Data Streams

- **Additional named $DATA attributes in a file's MFT entry**
- **Each can point to an unique set of cluster runs**
- **All the data streams share the same Standard Information and Filename attributes**
  - **So they all share the same timestamps**

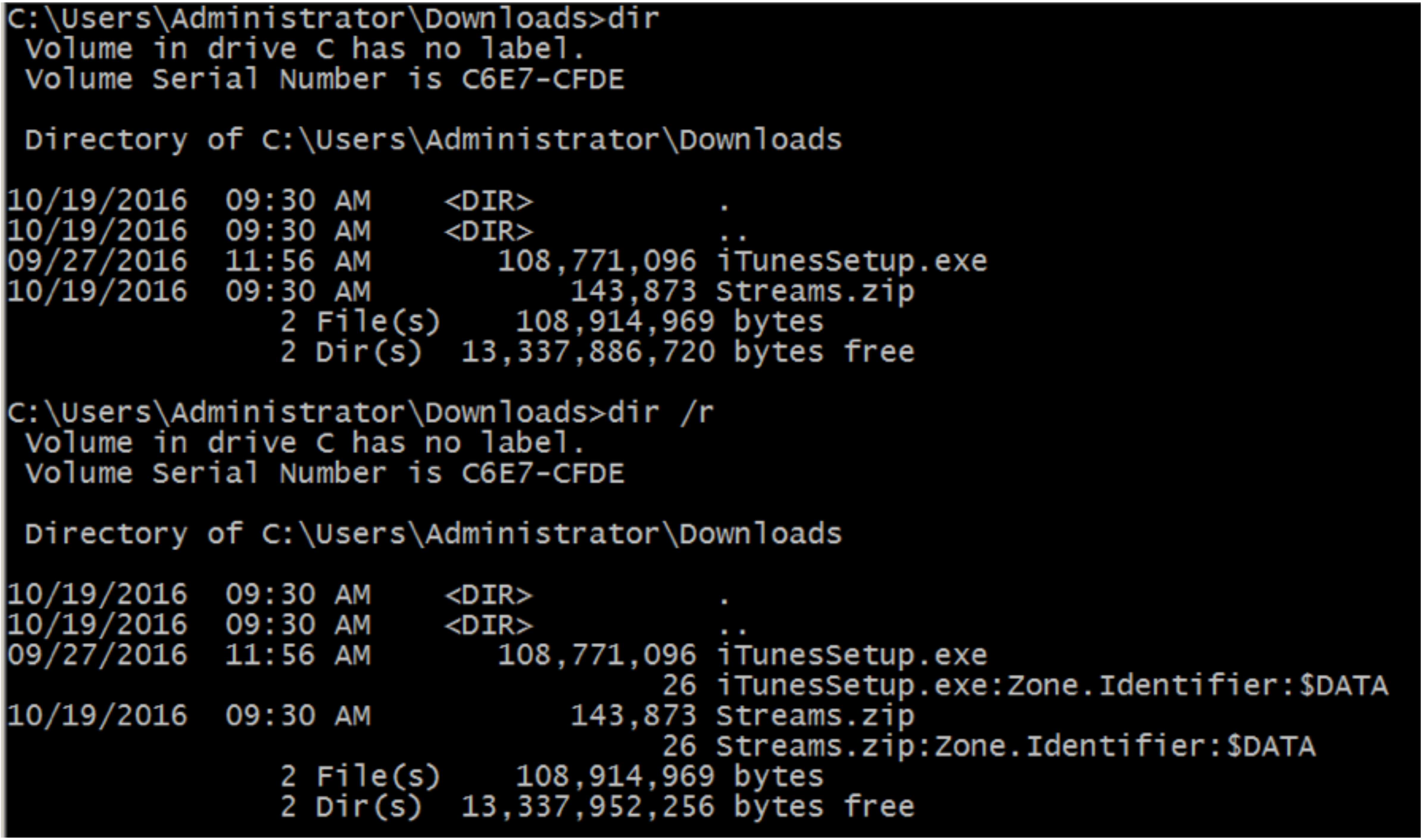


```
c:\test>echo "Hello world" > out.txt

c:\test>echo "I'm an ADS" > out.txt:secret.txt

c:\test>dir out.txt
 Volume in drive C is OSDisk
 Volume Serial Number is D681-E283

 Directory of c:\test

05/18/2014  02:28 PM                16 out.txt
               1 File(s)             16 bytes
               0 Dir(s)  105,934,098,432 bytes free

c:\test>dir /r
 Volume in drive C is OSDisk
 Volume Serial Number is D681-E283

 Directory of c:\test

05/18/2014  02:28 PM    <DIR>          .
05/18/2014  02:28 PM    <DIR>          ..
05/18/2014  02:28 PM                16 out.txt
                                    15 out.txt:secret.txt:$DATA
               1 File(s)             16 bytes
               2 Dir(s)  105,934,098,432 bytes free

c:\test>more < out.txt
"Hello world"

c:\test>more < out.txt:secret.txt
"I'm an ADS"
```

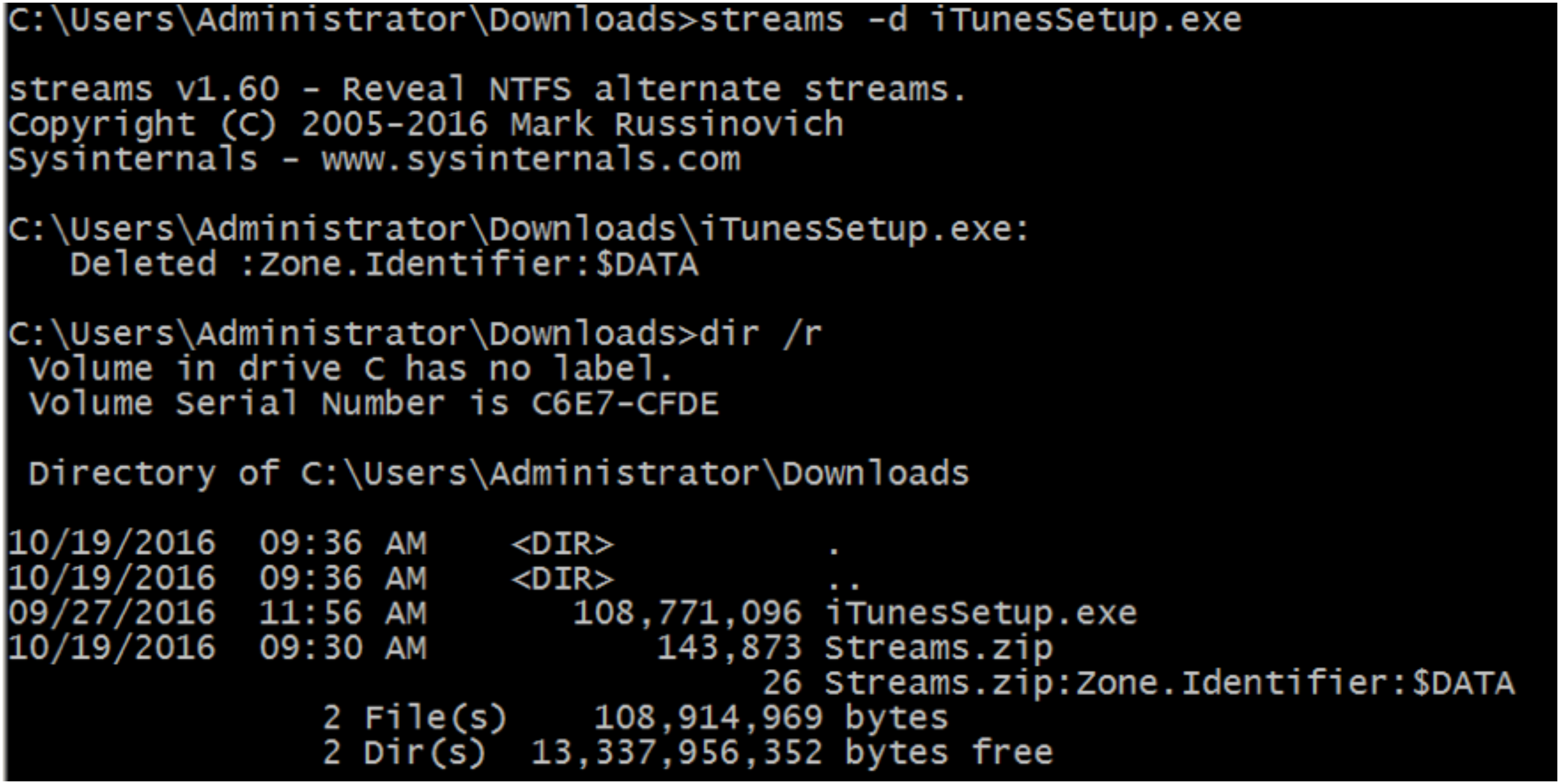
Figure 12-4. Creation and display of an alternate data stream

Known Alternate Stream Names

- **Browsers append a stream to downloaded files**
  - **Named Zone.Identifier**
- **Windows Explorer uses this data to determine the origin of a file and enforce security controls on it**
  - **Link Ch 12c**

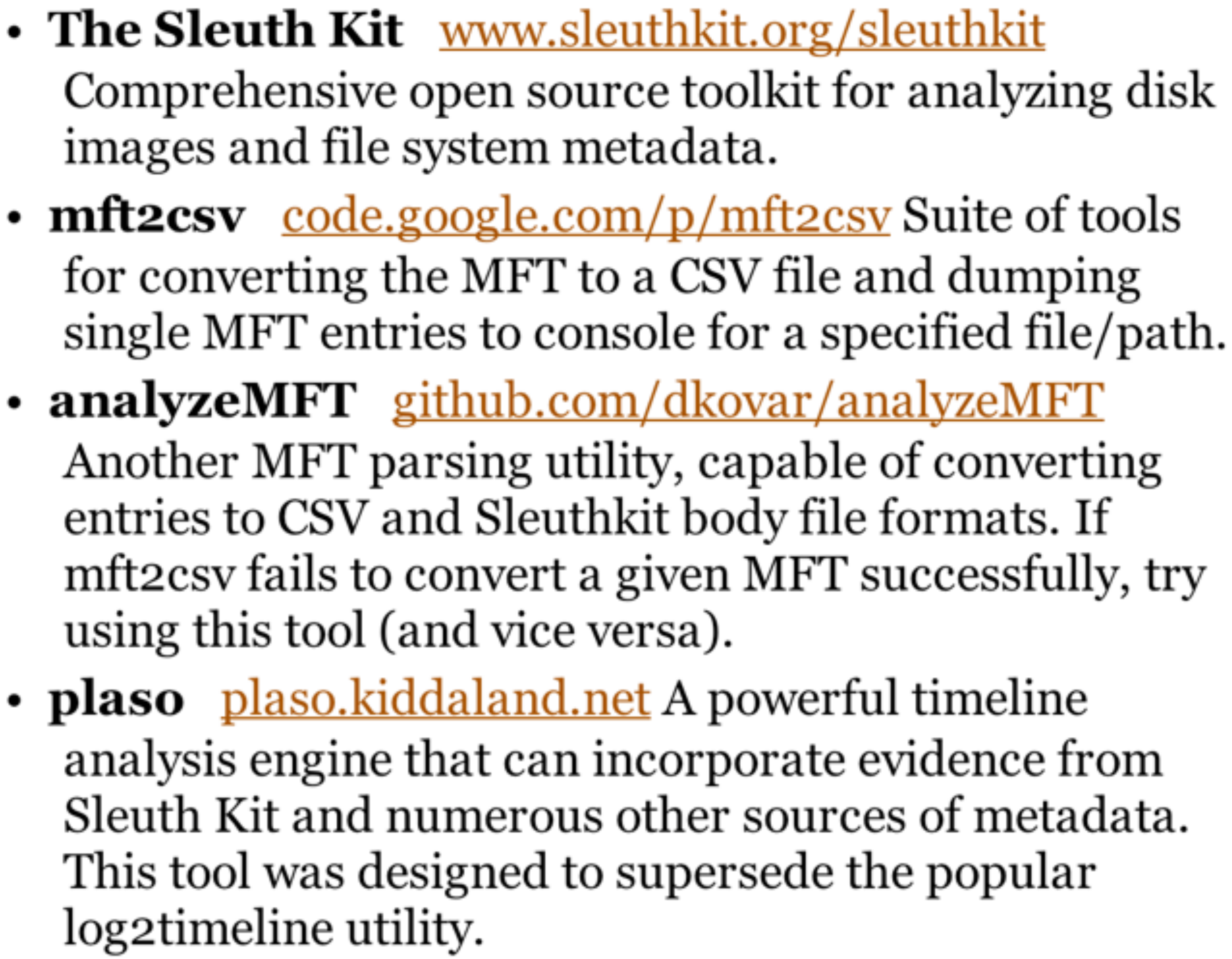


```
C:\Users\Administrator\Downloads>dir
 Volume in drive C has no label.
 Volume Serial Number is C6E7-CFDE

 Directory of C:\Users\Administrator\Downloads

10/19/2016  09:30 AM    <DIR>          .
10/19/2016  09:30 AM    <DIR>          ..
09/27/2016  11:56 AM       108,771,096 iTunesSetup.exe
10/19/2016  09:30 AM           143,873 Streams.zip
               2 File(s)    108,914,969 bytes
               2 Dir(s)  13,337,886,720 bytes free

C:\Users\Administrator\Downloads>dir /r
 Volume in drive C has no label.
 Volume Serial Number is C6E7-CFDE

 Directory of C:\Users\Administrator\Downloads

10/19/2016  09:30 AM    <DIR>          .
10/19/2016  09:30 AM    <DIR>          ..
09/27/2016  11:56 AM       108,771,096 iTunesSetup.exe
                                    26 iTunesSetup.exe:Zone.Identifier:$DATA
10/19/2016  09:30 AM           143,873 Streams.zip
                                    26 Streams.zip:Zone.Identifier:$DATA
               2 File(s)    108,914,969 bytes
               2 Dir(s)  13,337,952,256 bytes free
```



```
C:\Users\Administrator\Downloads>streams -d iTunesSetup.exe

streams v1.60 - Reveal NTFS alternate streams.
Copyright (C) 2005-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

C:\Users\Administrator\Downloads\iTunesSetup.exe:
   Deleted :Zone.Identifier:$DATA

C:\Users\Administrator\Downloads>dir /r
 Volume in drive C has no label.
 Volume Serial Number is C6E7-CFDE

 Directory of C:\Users\Administrator\Downloads

10/19/2016  09:36 AM    <DIR>          .
10/19/2016  09:36 AM    <DIR>          ..
09/27/2016  11:56 AM       108,771,096 iTunesSetup.exe
10/19/2016  09:30 AM           143,873 Streams.zip
                                    26 Streams.zip:Zone.Identifier:$DATA
               2 File(s)    108,914,969 bytes
               2 Dir(s)  13,337,956,352 bytes free
```

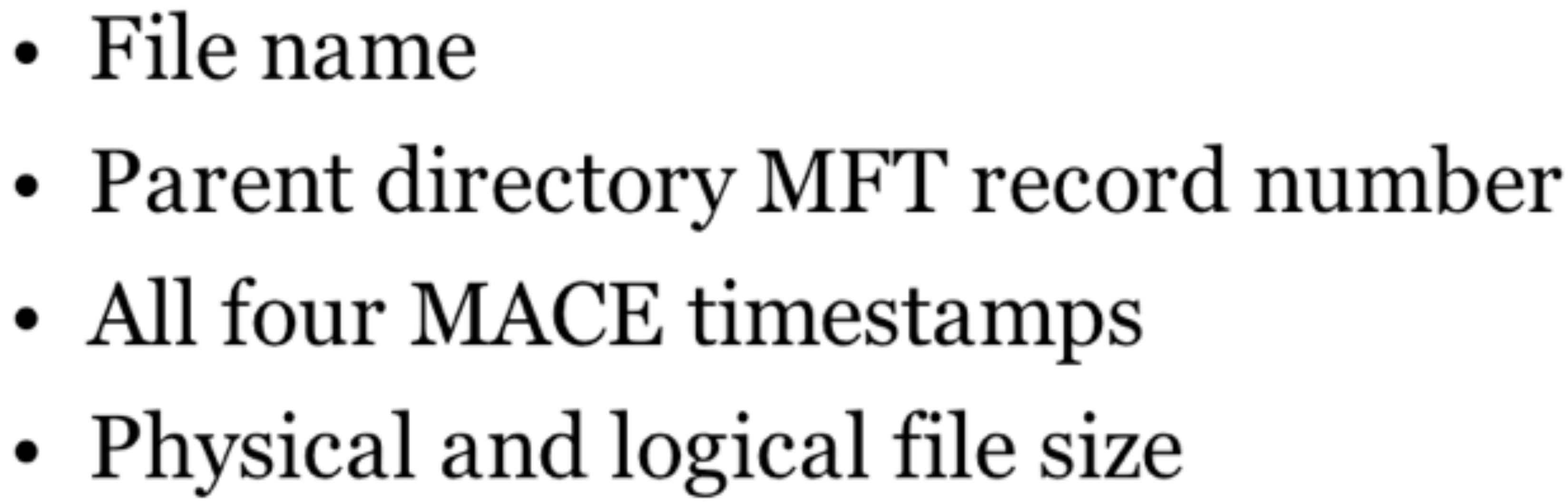
MFT Analysis Tools



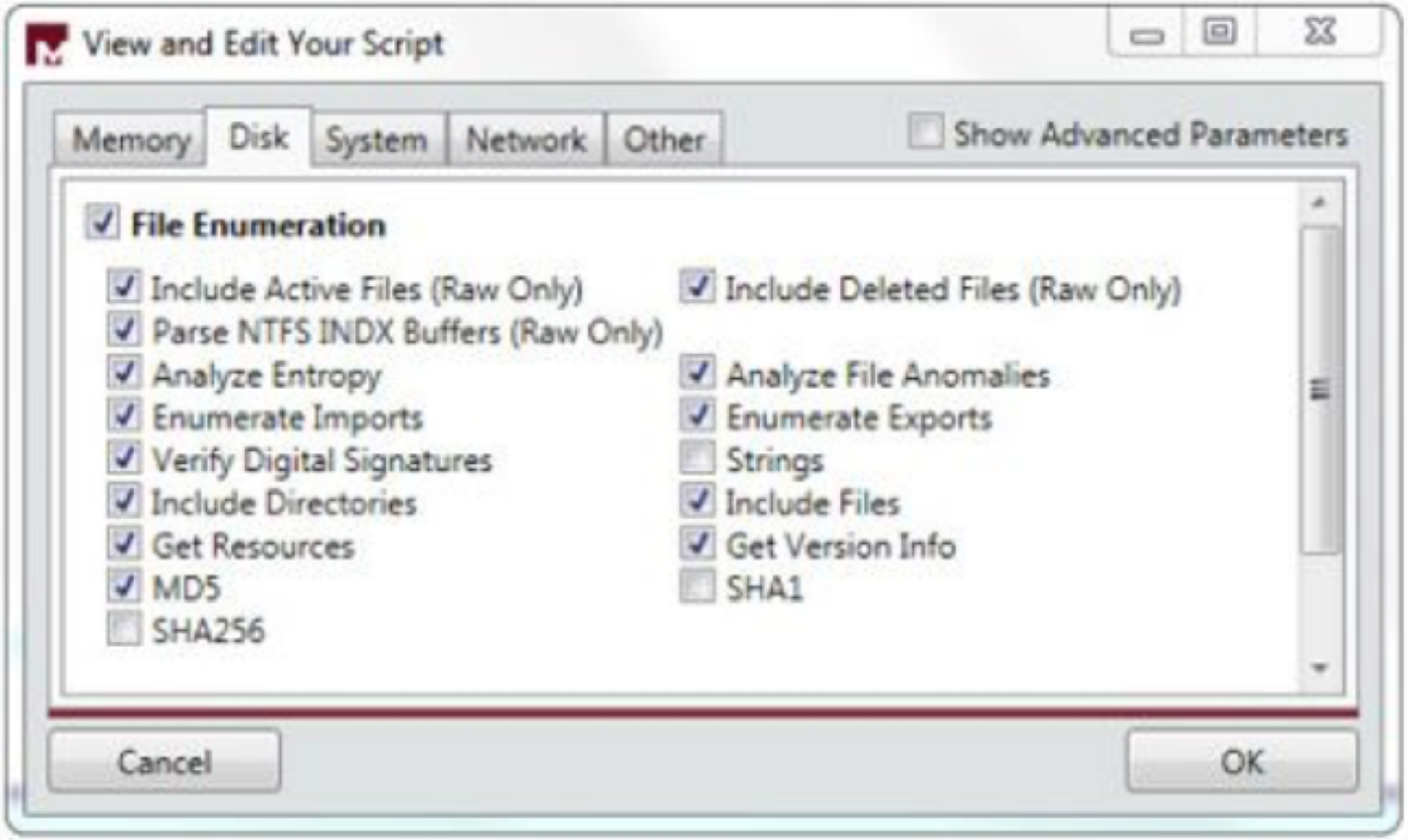
- **The Sleuth Kit** www.sleuthkit.org/sleuthkit Comprehensive open source toolkit for analyzing disk images and file system metadata.
- **mft2csv** code.google.com/p/mft2csv Suite of tools for converting the MFT to a CSV file and dumping single MFT entries to console for a specified file/path.
- **analyzeMFT** github.com/dkovar/analyzeMFT Another MFT parsing utility, capable of converting entries to CSV and Sleuthkit body file formats. If mft2csv fails to convert a given MFT successfully, try using this tool (and vice versa).
- **plaso** plaso.kiddaland.net A powerful timeline analysis engine that can incorporate evidence from Sleuth Kit and numerous other sources of metadata. This tool was designed to supersede the popular log2timeline utility.

INDX Attributes

- **Used to make file searches faster**
- **Often contains metadata from deleted files**
  - **Links Ch 12h, 12i**



- File name
- Parent directory MFT record number
- All four MACE timestamps
- Physical and logical file size



□ Show Advanced Parameters

☑ **File Enumeration**

☑ Include Active Files (Raw Only)
☑ Parse NTFS INDX Buffers (Raw Only)
☑ Analyze Entropy
☑ Enumerate Imports
☑ Verify Digital Signatures
☑ Include Directories
☑ Get Resources
☑ MD5
□ SHA256

☑ Include Deleted Files (Raw Only)
☑ Analyze File Anomalies
☑ Enumerate Exports
□ Strings
☑ Include Files
☑ Get Version Info
□ SHA1

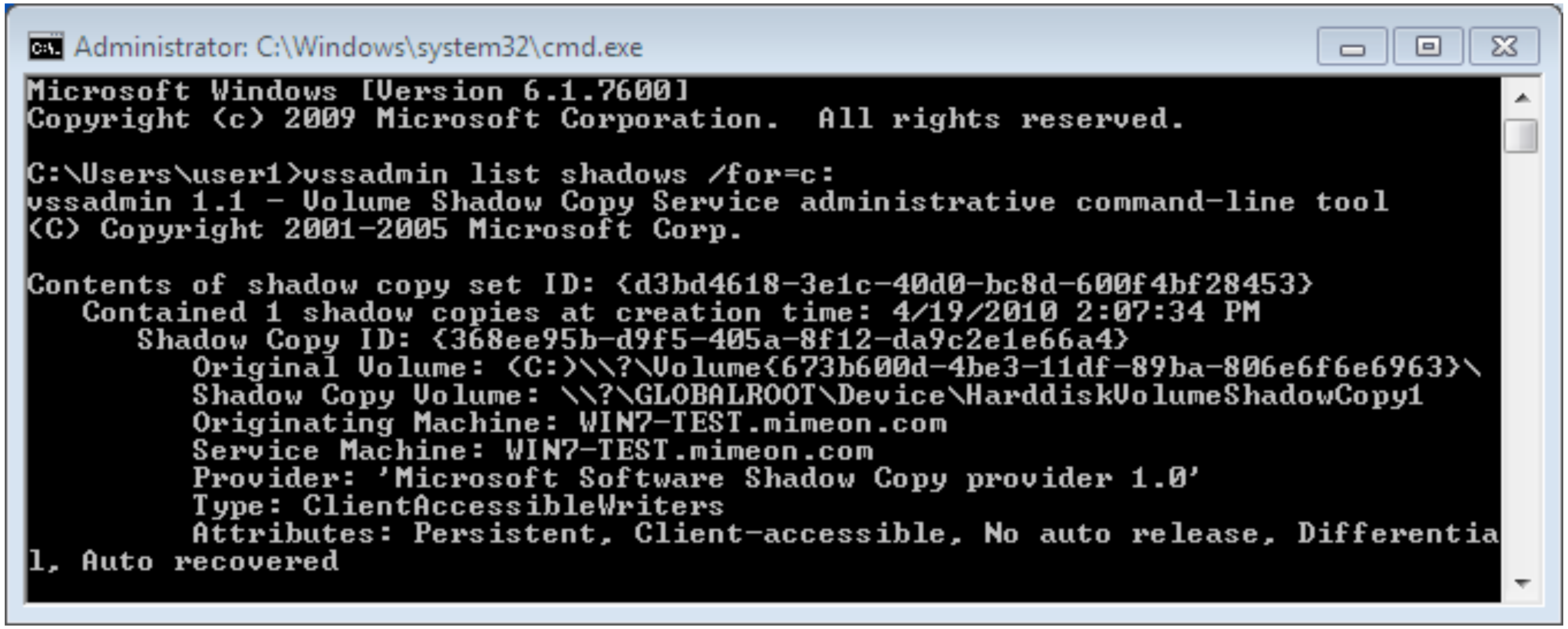
Figure 12-6. Script collector option in Redline to enable INDX parsing

Change Logs

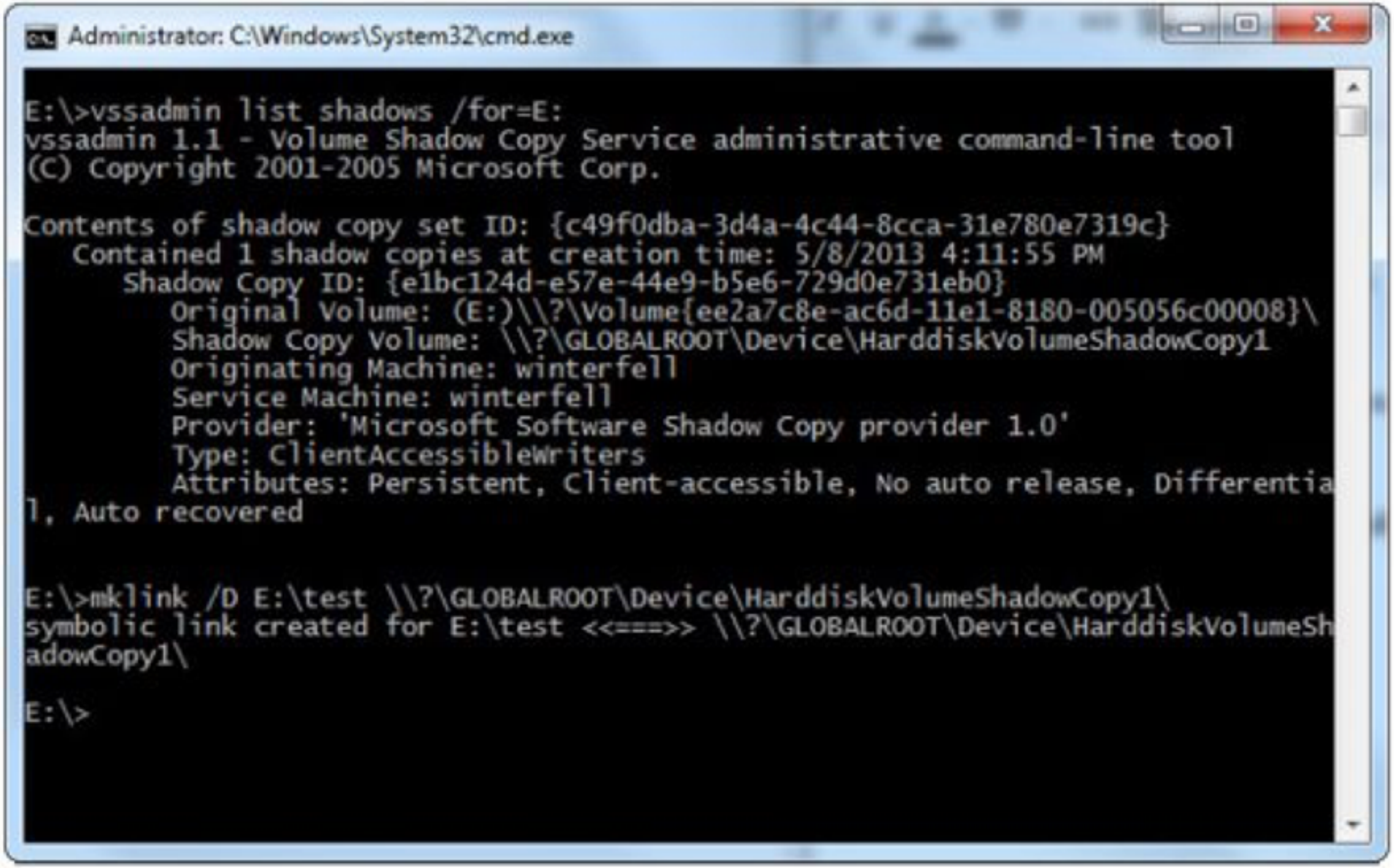
- **$LogFile tracks all transactions that change the structure of a volume**
  - **File or directory creation/copy/delete**
  - **Changes to file metadata or INDX records**
- **$UsnJrnl (Update Sequence Number) journal**
  - **Tracks less data but has a longer history**

Volume Shadow Copies

- **Automatically created backup of Windows files**
- **Manage with the vssadmin and mklink command-line tools (link Ch 12k)**



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\user1>vssadmin list shadows /for=c:
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2005 Microsoft Corp.

Contents of shadow copy set ID: {d3bd4618-3e1c-40d0-bc8d-600f4bf28453}
   Contained 1 shadow copies at creation time: 4/19/2010 2:07:34 PM
      Shadow Copy ID: {368ee95b-d9f5-405a-8f12-da9c2e1e66a4}
         Original Volume: (C:)\\?\Volume{673b600d-4be3-11df-89ba-806e6f6e6963}\
         Shadow Copy Volume: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1
         Originating Machine: WIN7-TEST.mimeon.com
         Service Machine: WIN7-TEST.mimeon.com
         Provider: 'Microsoft Software Shadow Copy provider 1.0'
         Type: ClientAccessibleWriters
         Attributes: Persistent, Client-accessible, No auto release, Differentia
l, Auto recovered
```

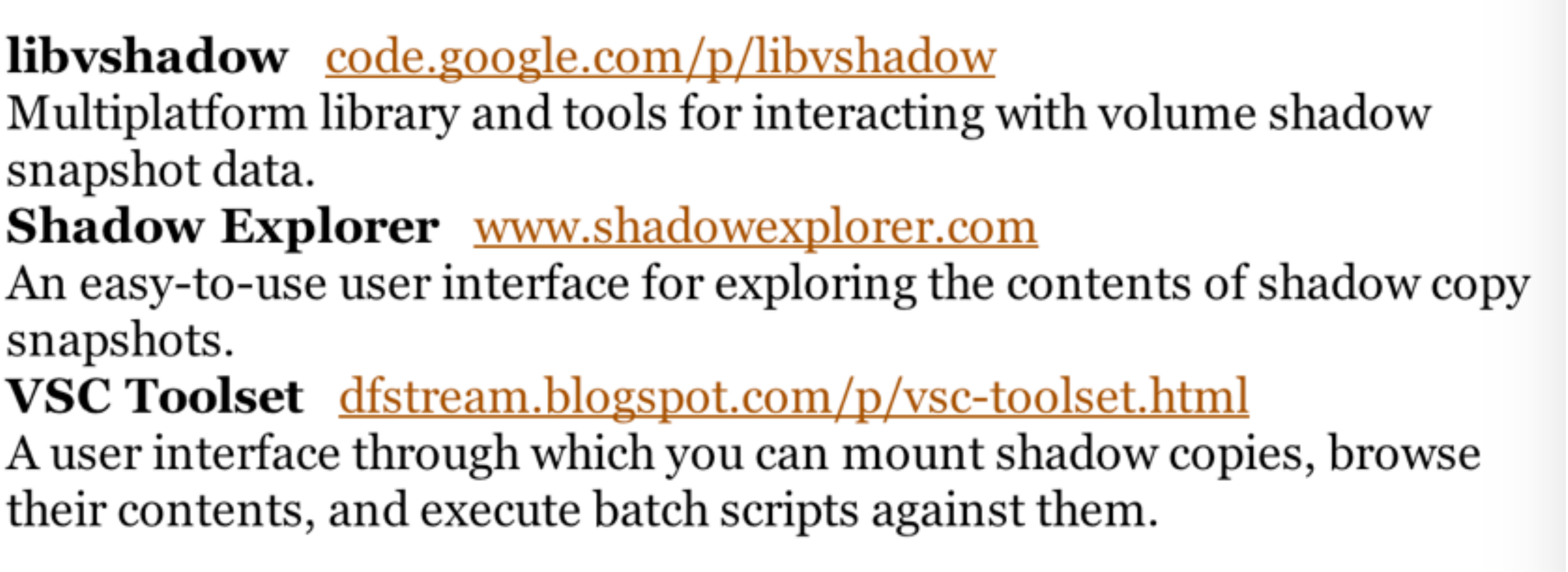


```
E:\>vssadmin list shadows /for=E:
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2005 Microsoft Corp.

Contents of shadow copy set ID: {c49f0dba-3d4a-4c44-8cca-31e780e7319c}
   Contained 1 shadow copies at creation time: 5/8/2013 4:11:55 PM
      Shadow Copy ID: {e1bc124d-e57e-44e9-b5e6-729d0e731eb0}
         Original Volume: (E:)\\?\Volume{ee2a7c8e-ac6d-11e1-8180-005056c00008}\
         Shadow Copy Volume: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1
         Originating Machine: winterfell
         Service Machine: winterfell
         Provider: 'Microsoft Software Shadow Copy provider 1.0'
         Type: ClientAccessibleWriters
         Attributes: Persistent, Client-accessible, No auto release, Differentia
l, Auto recovered


E:\>mklink /D E:\test \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\
symbolic link created for E:\test <<===>> \\?\GLOBALROOT\Device\HarddiskVolumeSh
adowCopy1\

E:\>
```

Figure 12-7. Syntax to list shadow copies for a volume and mount via symbolic link

Shadow Copy

- **A mirror of the volume's entire file system at the time of the snapshot**
- **Available within the linked directory**
- **Other tools:**



libvshadow code.google.com/p/libvshadow
Multiplatform library and tools for interacting with volume shadow snapshot data.

Shadow Explorer www.shadowexplorer.com
An easy-to-use user interface for exploring the contents of shadow copy snapshots.

VSC Toolset dfstream.blogspot.com/p/vsc-toolset.html
A user interface through which you can mount shadow copies, browse their contents, and execute batch scripts against them.

File System Redirector

- **Windows 32-bit on Windows 64-bit (WoW64)**
- **Redirects some folders elsewhere when 32-bit programs run on 64-bit Windows, like**
- **%SYSTEMROOT%\system32 redirects to C:\Windows\SysWOW64**
- **32-bit tools may not see the whole file system**

Windows Prefetch

C:\Windows\Prefetch Contains

- **NTOSBOOT-BooDFAAD.pf (system boot prefetch) -- only file existing on Windows Server by default**

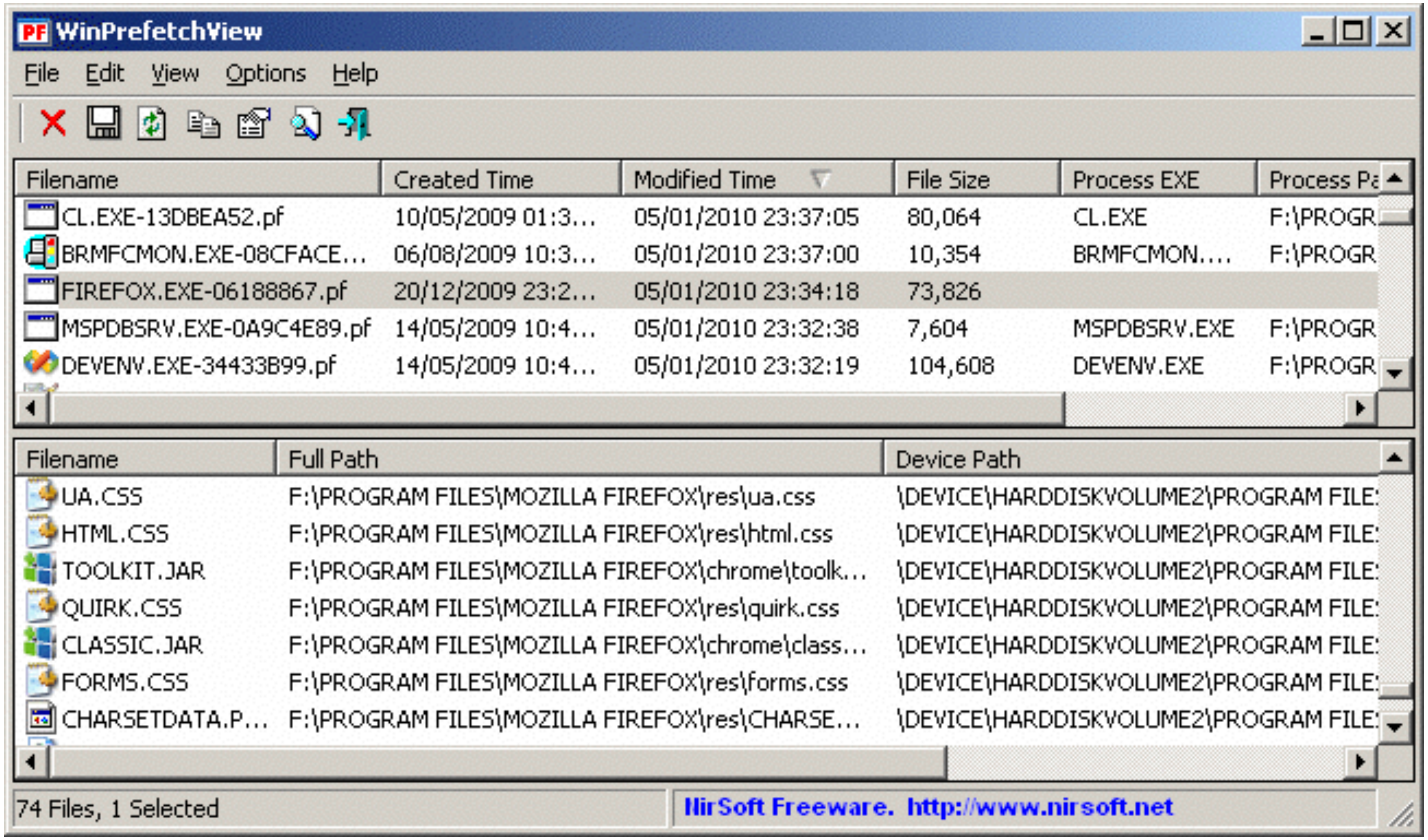
- **Layout.ini (for disk defragmenter)**

- **Appname-########.pf (up to 128 application-specific prefetch files)**

Value

- **A record of programs executed on a system**
- **Even if the executable has been deleted**
- **Shows when application was first run, when it most recently ran, and how many times it was run**
- **Also shows each component loaded**

WinPrefetchView



Filename	Created Time	Modified Time	File Size	Process EXE	Process Pa
CL.EXE-13DBEA52.pf	10/05/2009 01:3...	05/01/2010 23:37:05	80,064	CL.EXE	F:\PROGR
BRMFCMON.EXE-08CFACE...	06/08/2009 10:3...	05/01/2010 23:37:00	10,354	BRMFCMON....	F:\PROGR
FIREFOX.EXE-06188867.pf	20/12/2009 23:2...	05/01/2010 23:34:18	73,826		
MSPDBSRV.EXE-0A9C4E89.pf	14/05/2009 10:4...	05/01/2010 23:32:38	7,604	MSPDBSRV.EXE	F:\PROGR
DEVENV.EXE-34433B99.pf	14/05/2009 10:4...	05/01/2010 23:32:19	104,608	DEVENV.EXE	F:\PROGR

Filename	Full Path	Device Path
UA.CSS	F:\PROGRAM FILES\MOZILLA FIREFOX\res\ua.css	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE
HTML.CSS	F:\PROGRAM FILES\MOZILLA FIREFOX\res\html.css	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE
TOOLKIT.JAR	F:\PROGRAM FILES\MOZILLA FIREFOX\chrome\toolk...	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE
QUIRK.CSS	F:\PROGRAM FILES\MOZILLA FIREFOX\res\quirk.css	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE
CLASSIC.JAR	F:\PROGRAM FILES\MOZILLA FIREFOX\chrome\class...	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE
FORMS.CSS	F:\PROGRAM FILES\MOZILLA FIREFOX\res\forms.css	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE
CHARSETDATA.P...	F:\PROGRAM FILES\MOZILLA FIREFOX\res\CHARSE...	\DEVICE\HARDDISKVOLUME2\PROGRAM FILE

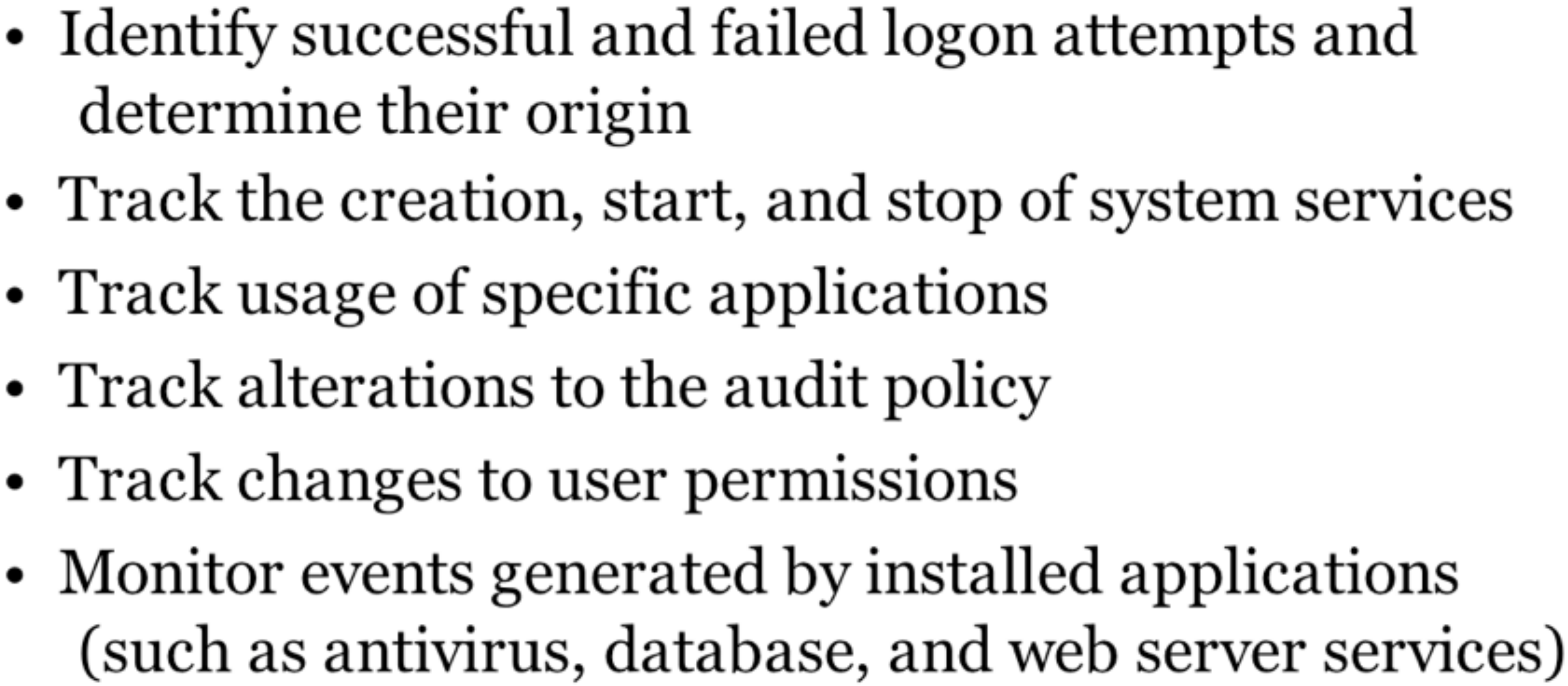
74 Files, 1 Selected

NirSoft Freeware. http://www.nirsoft.net

- **Link Ch 12l**

Event Logs

Event Logs Enable these Tasks



- Identify successful and failed logon attempts and determine their origin
- Track the creation, start, and stop of system services
- Track usage of specific applications
- Track alterations to the audit policy
- Track changes to user permissions
- Monitor events generated by installed applications (such as antivirus, database, and web server services)

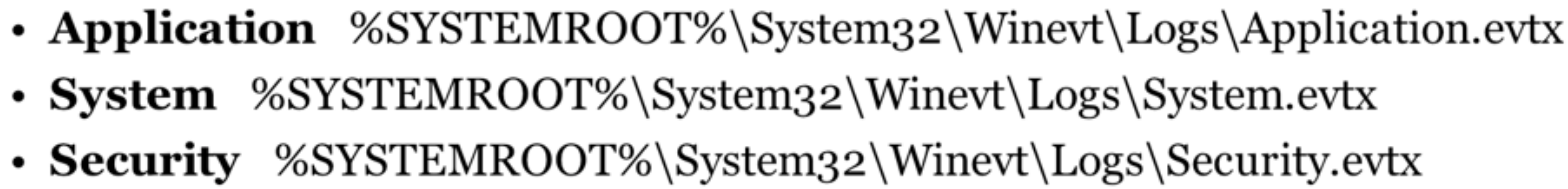
Types of Logs

- **Core event logs in all Windows versions**
  - **Application**
    - **Errors and info from apps; antivirus and host-based IPS logs**
  - **System**
    - **Events from core Windows services; changes in time, driver loads, network configuration issues**
  - **Security**
    - **Login and logoff attempts, changes to audit policy**

Acquiring Logs

- **Log file locations are specified in this Registry key: HKLM\SYSTEM\CurrentControlSet\Services\Eventlog**

- **For Vista and later, the logs are in these XML files:**



- **Application** %SYSTEMROOT%\System32\Winevt\Logs\Application.evtx
- **System** %SYSTEMROOT%\System32\Winevt\Logs\System.evtx
- **Security** %SYSTEMROOT%\System32\Winevt\Logs\Security.evtx

Applications and Services Logs

- **EVTX files in %SYSTEMROOT%\System32\Winevt\Logs\**
- **Logs for Task scheduler, Windows Firewall, AppLocker, Terminal Services, User Access Control**

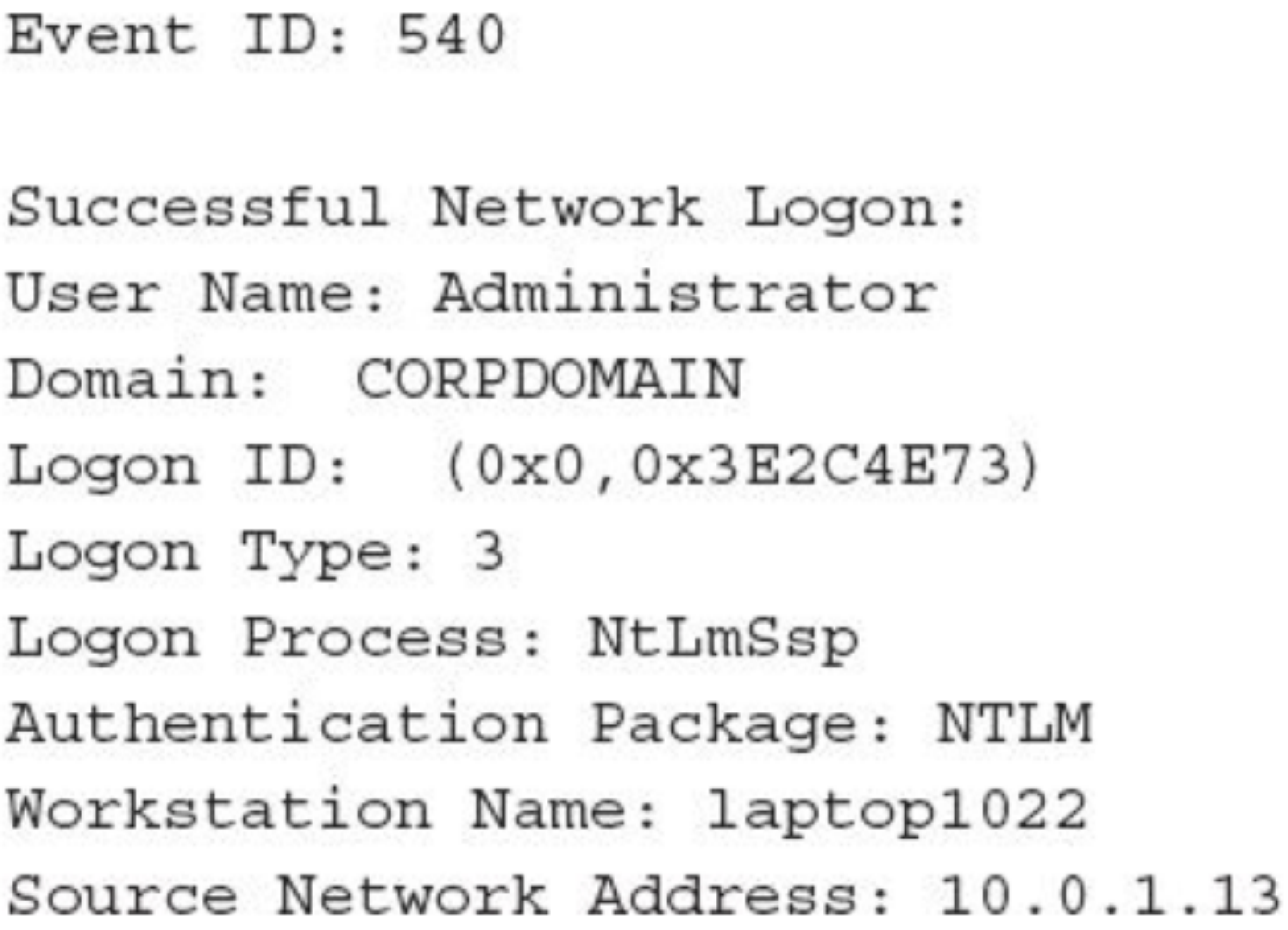
Event ID

- **Each event is labelled with its Source and Event ID number**

- **Good resource: eventid.net**

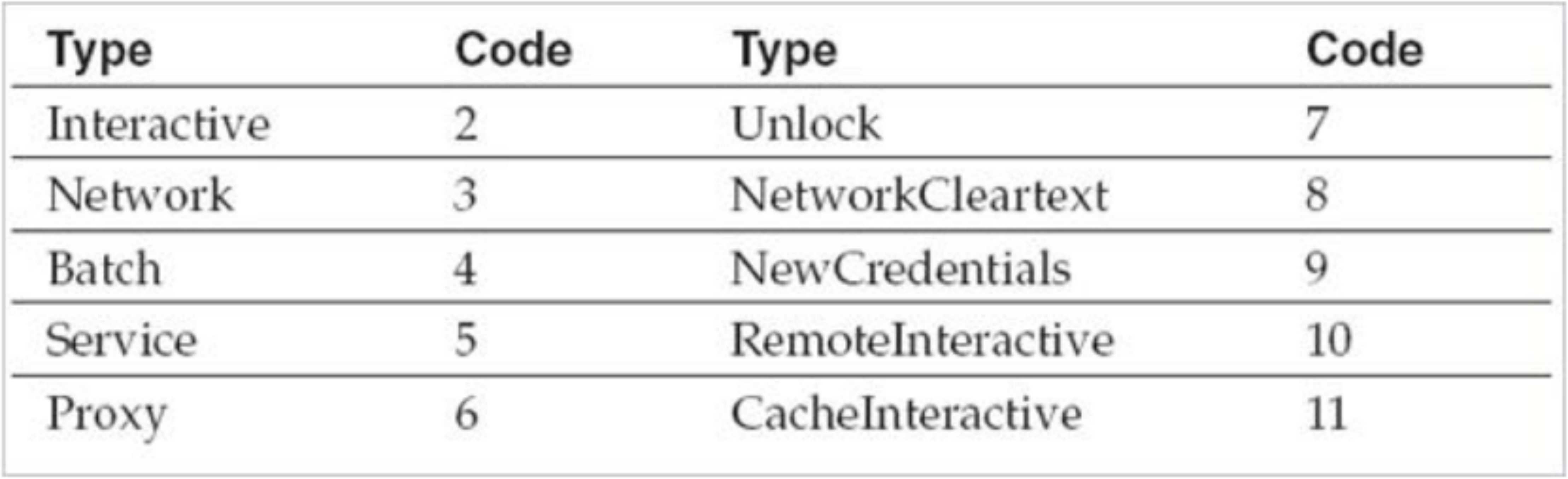
- **Vista and later often have EventIDs that are 4096 larger than the EventID from Windows XP**

Logon Events



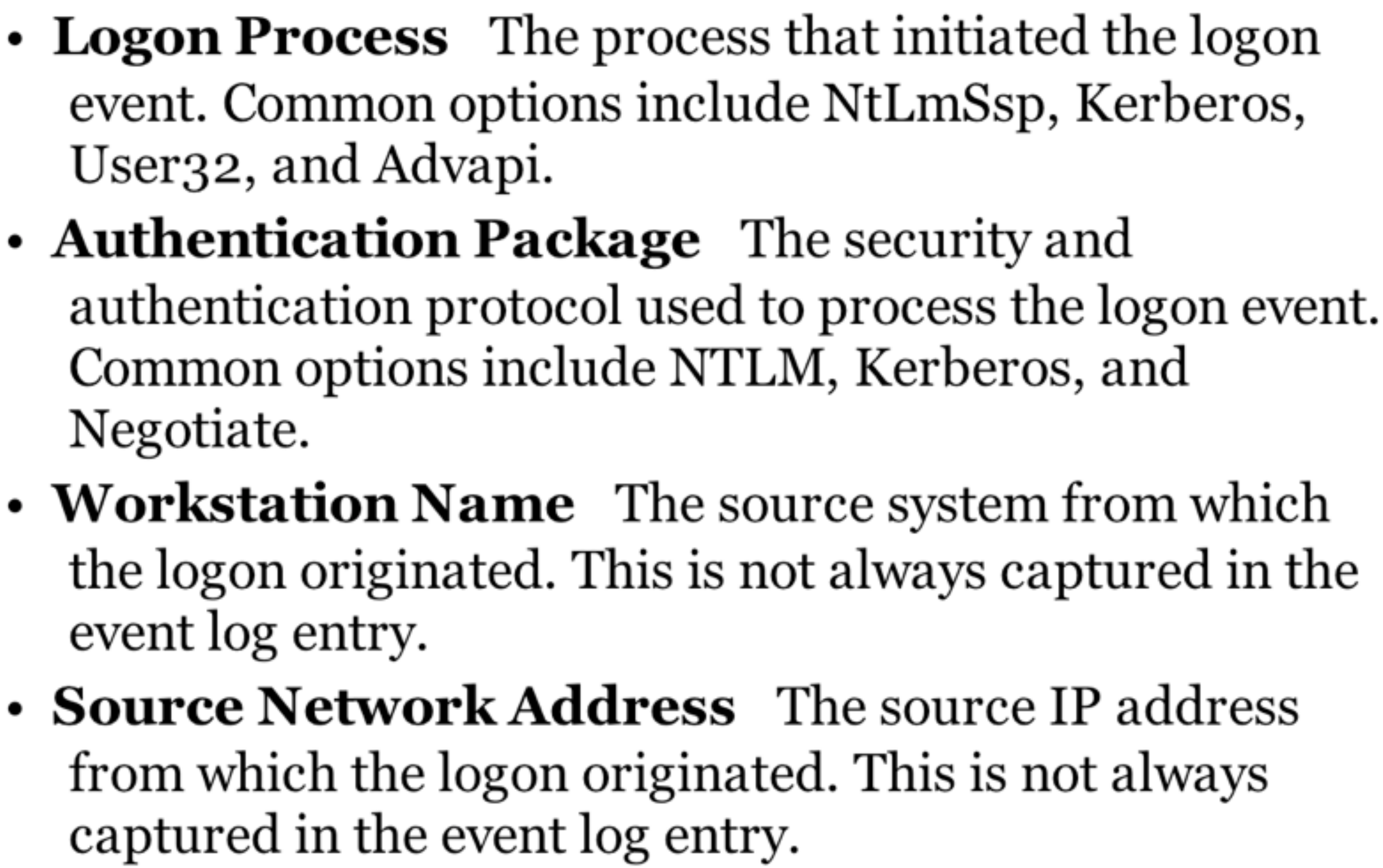
```
Event ID: 540

Successful Network Logon:
User Name: Administrator
Domain:  CORPDOMAIN
Logon ID:  (0x0,0x3E2C4E73)
Logon Type: 3
Logon Process: NtLmSsp
Authentication Package: NTLM
Workstation Name: laptop1022
Source Network Address: 10.0.1.13
```

- **Logon ID** A unique session identifier. You can use this value as a search term or filter to find all event log entries associated with this specific logon session.
- **Logon Type** A code referencing the type of logon initiated by the user. The following table provides further detail on the Logon Type field and its possible values:



Type	Code	Type	Code
Interactive	2	Unlock	7
Network	3	NetworkCleartext	8
Batch	4	NewCredentials	9
Service	5	RemoteInteractive	10
Proxy	6	CacheInteractive	11

Fields

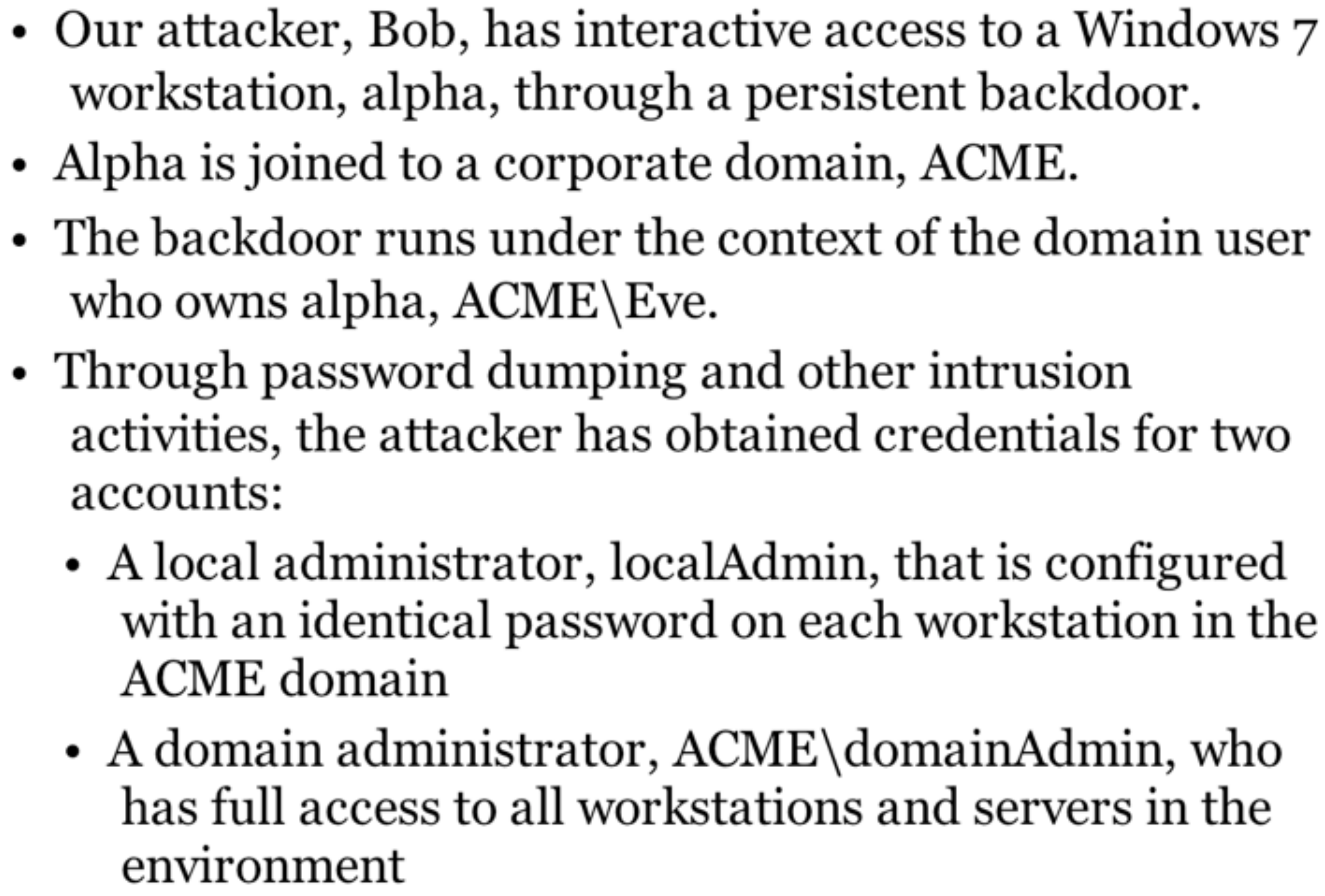


- **Logon Process** The process that initiated the logon event. Common options include NtLmSsp, Kerberos, User32, and Advapi.
- **Authentication Package** The security and authentication protocol used to process the logon event. Common options include NTLM, Kerberos, and Negotiate.
- **Workstation Name** The source system from which the logon originated. This is not always captured in the event log entry.
- **Source Network Address** The source IP address from which the logon originated. This is not always captured in the event log entry.

Lateral Movement

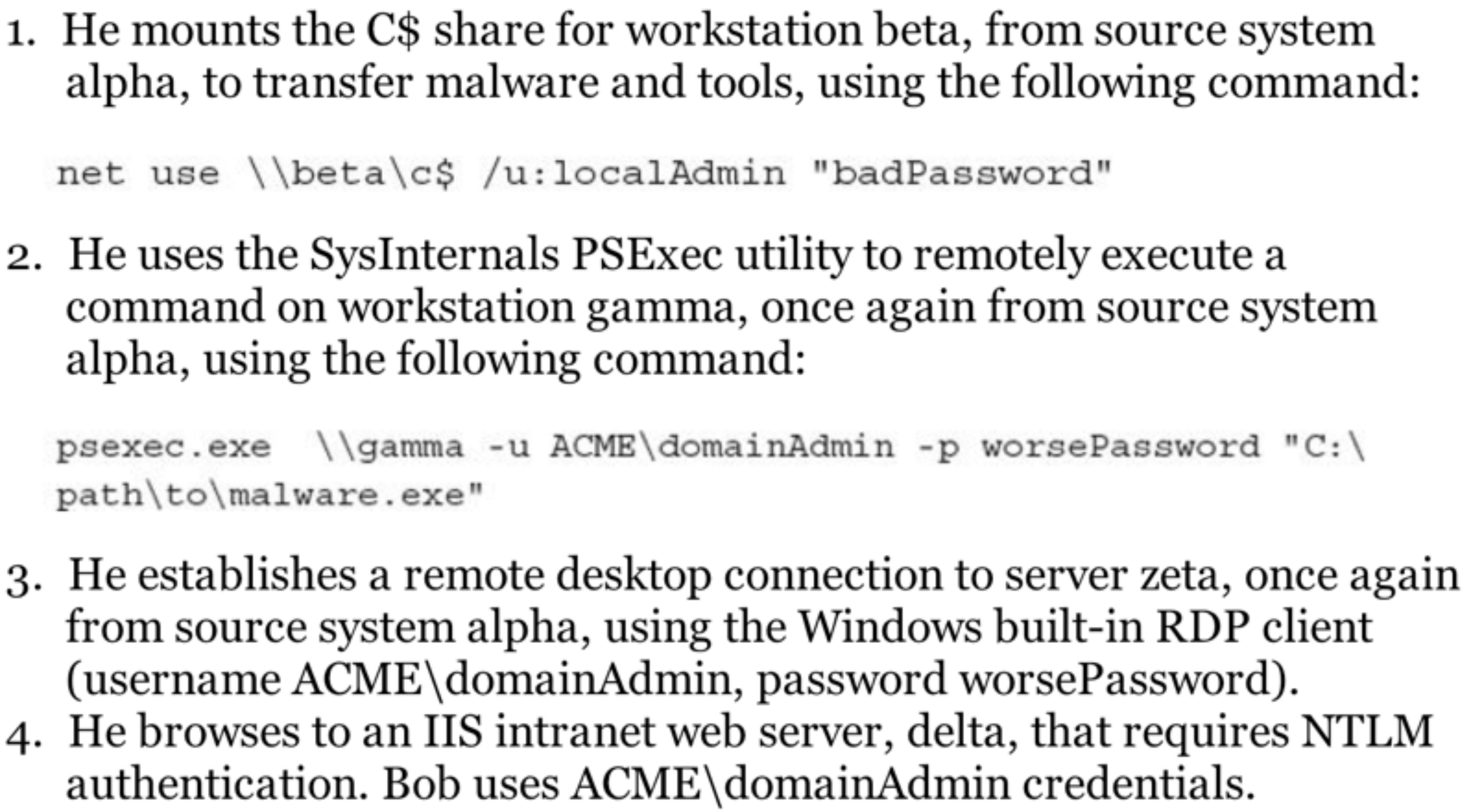
- **Attackers use stolen credentials to move from system to system**
- **Often use a common administrator account**
- **Or a domain or domain administrator account**

Example



- Our attacker, Bob, has interactive access to a Windows 7 workstation, alpha, through a persistent backdoor.
- Alpha is joined to a corporate domain, ACME.
- The backdoor runs under the context of the domain user who owns alpha, ACME\Eve.
- Through password dumping and other intrusion activities, the attacker has obtained credentials for two accounts:
  - A local administrator, localAdmin, that is configured with an identical password on each workstation in the ACME domain
  - A domain administrator, ACME\domainAdmin, who has full access to all workstations and servers in the environment

In Command Shell as ACME\Eve



1. He mounts the C$ share for workstation beta, from source system alpha, to transfer malware and tools, using the following command:

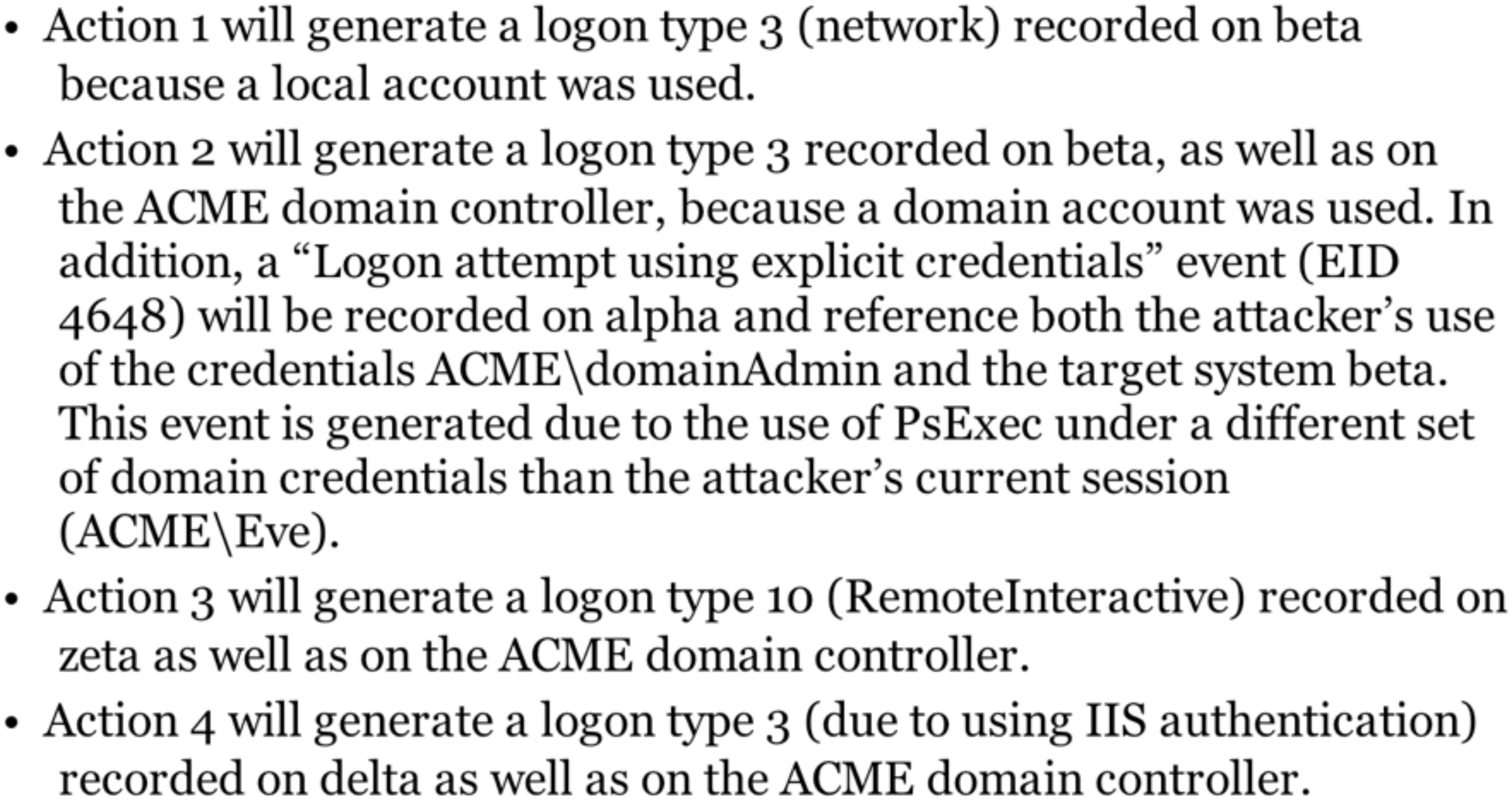
```
net use \\beta\c$ /u:localAdmin "badPassword"
```

2. He uses the SysInternals PSExec utility to remotely execute a command on workstation gamma, once again from source system alpha, using the following command:

```
psexec.exe  \\gamma -u ACME\domainAdmin -p worsePassword "C:\
path\to\malware.exe"
```

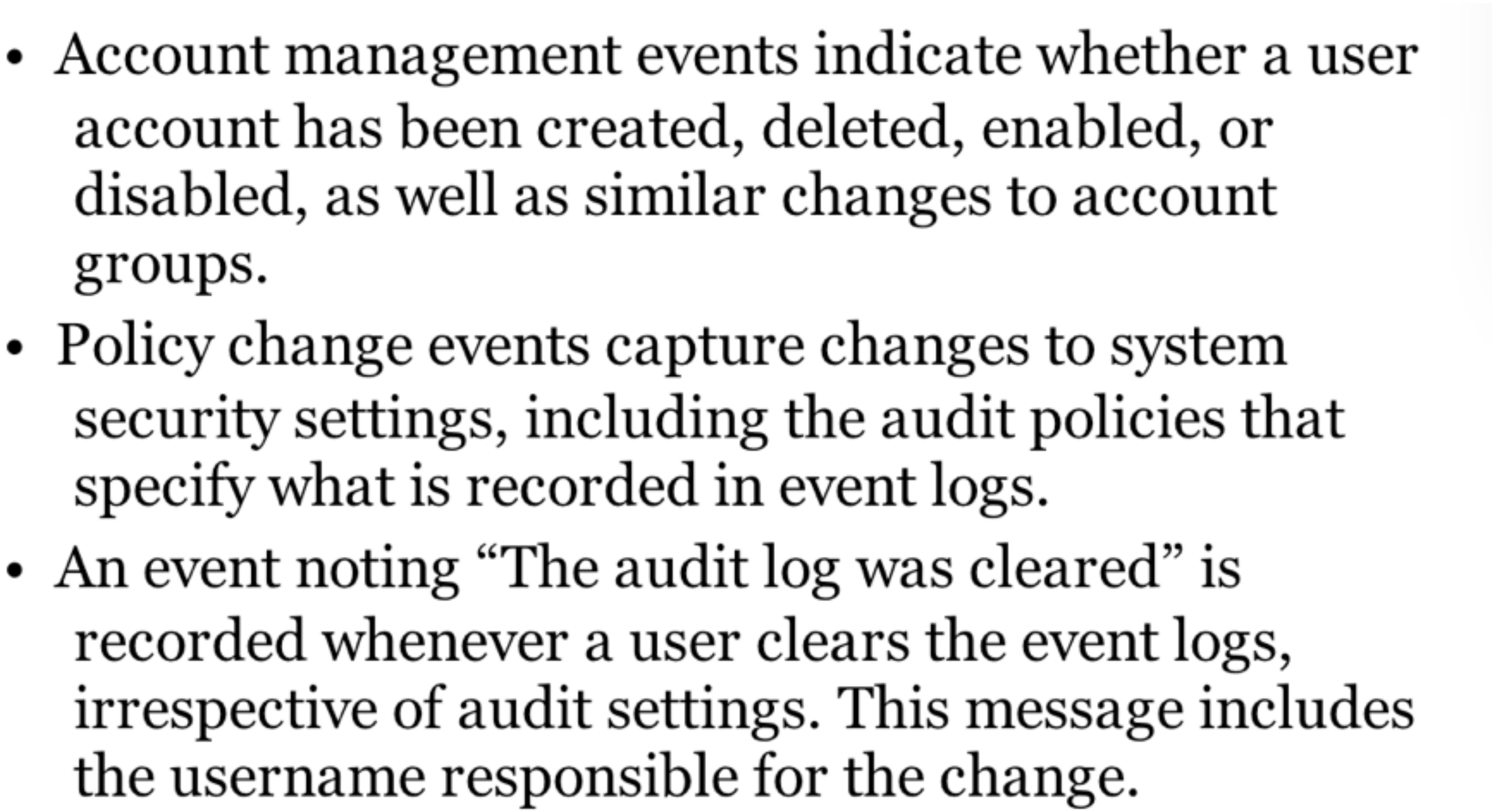
3. He establishes a remote desktop connection to server zeta, once again from source system alpha, using the Windows built-in RDP client (username ACME\domainAdmin, password worsePassword).
4. He browses to an IIS intranet web server, delta, that requires NTLM authentication. Bob uses ACME\domainAdmin credentials.

Events Logged



- Action 1 will generate a logon type 3 (network) recorded on beta because a local account was used.
- Action 2 will generate a logon type 3 recorded on beta, as well as on the ACME domain controller, because a domain account was used. In addition, a “Logon attempt using explicit credentials” event (EID 4648) will be recorded on alpha and reference both the attacker’s use of the credentials ACME\domainAdmin and the target system beta. This event is generated due to the use of PsExec under a different set of domain credentials than the attacker’s current session (ACME\Eve).
- Action 3 will generate a logon type 10 (RemoteInteractive) recorded on zeta as well as on the ACME domain controller.
- Action 4 will generate a logon type 3 (due to using IIS authentication) recorded on delta as well as on the ACME domain controller.

Changes to Accounts and Security Settings: Security Logs



- Account management events indicate whether a user account has been created, deleted, enabled, or disabled, as well as similar changes to account groups.
- Policy change events capture changes to system security settings, including the audit policies that specify what is recorded in event logs.
- An event noting “The audit log was cleared” is recorded whenever a user clears the event logs, irrespective of audit settings. This message includes the username responsible for the change.

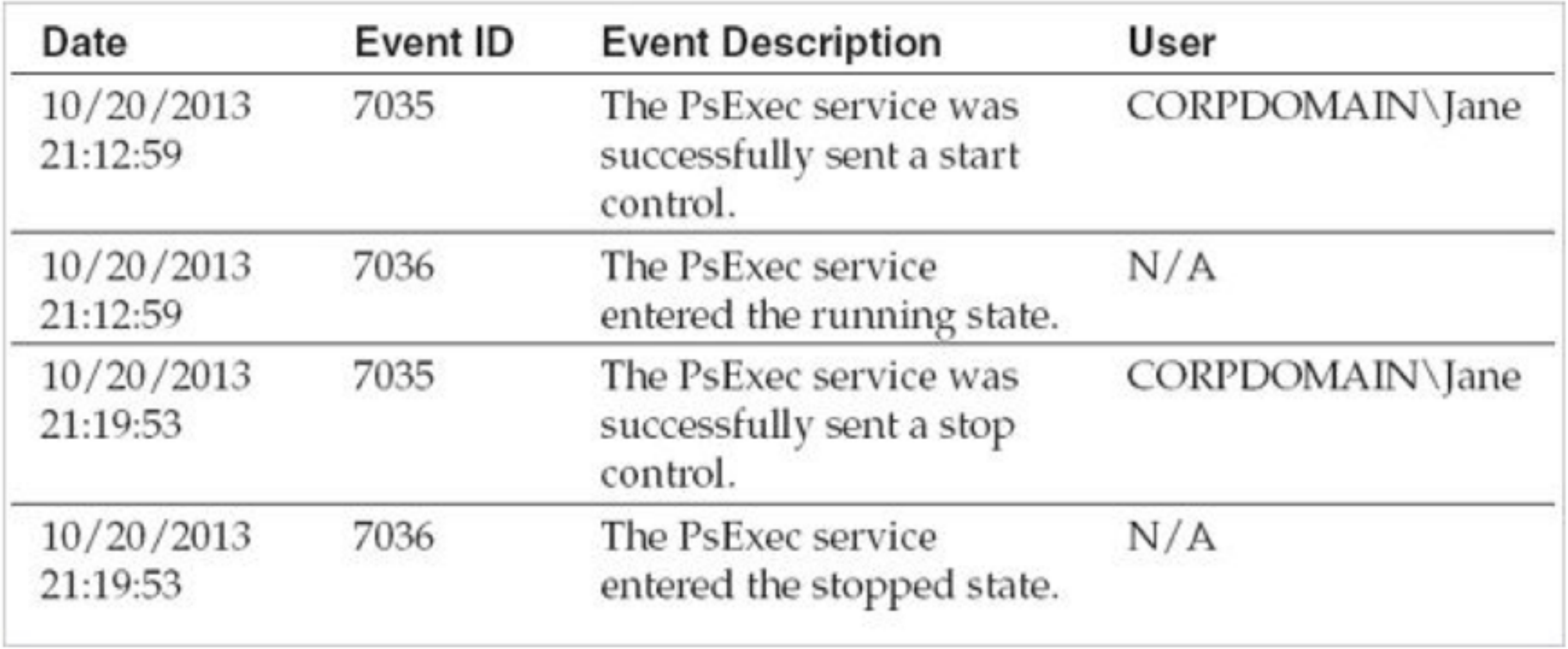
Process Auditing

- **Not on by default**
- **Turn it on in local audit policy or Group Policy**
- **Puts an event in the Security log every time a process is executed or terminated**
- **Generates a lot of log events**

Service Events

- **System logs record every time a service starts or stops**
- **A common persistence mechanism for malware**

Logs for PsExec



Date	Event ID	Event Description	User
10/20/2013 21:12:59	7035	The PsExec service was successfully sent a start control.	CORPDOMAIN\Jane
10/20/2013 21:12:59	7036	The PsExec service entered the running state.	N/A
10/20/2013 21:19:53	7035	The PsExec service was successfully sent a stop control.	CORPDOMAIN\Jane
10/20/2013 21:19:53	7036	The PsExec service entered the stopped state.	N/A

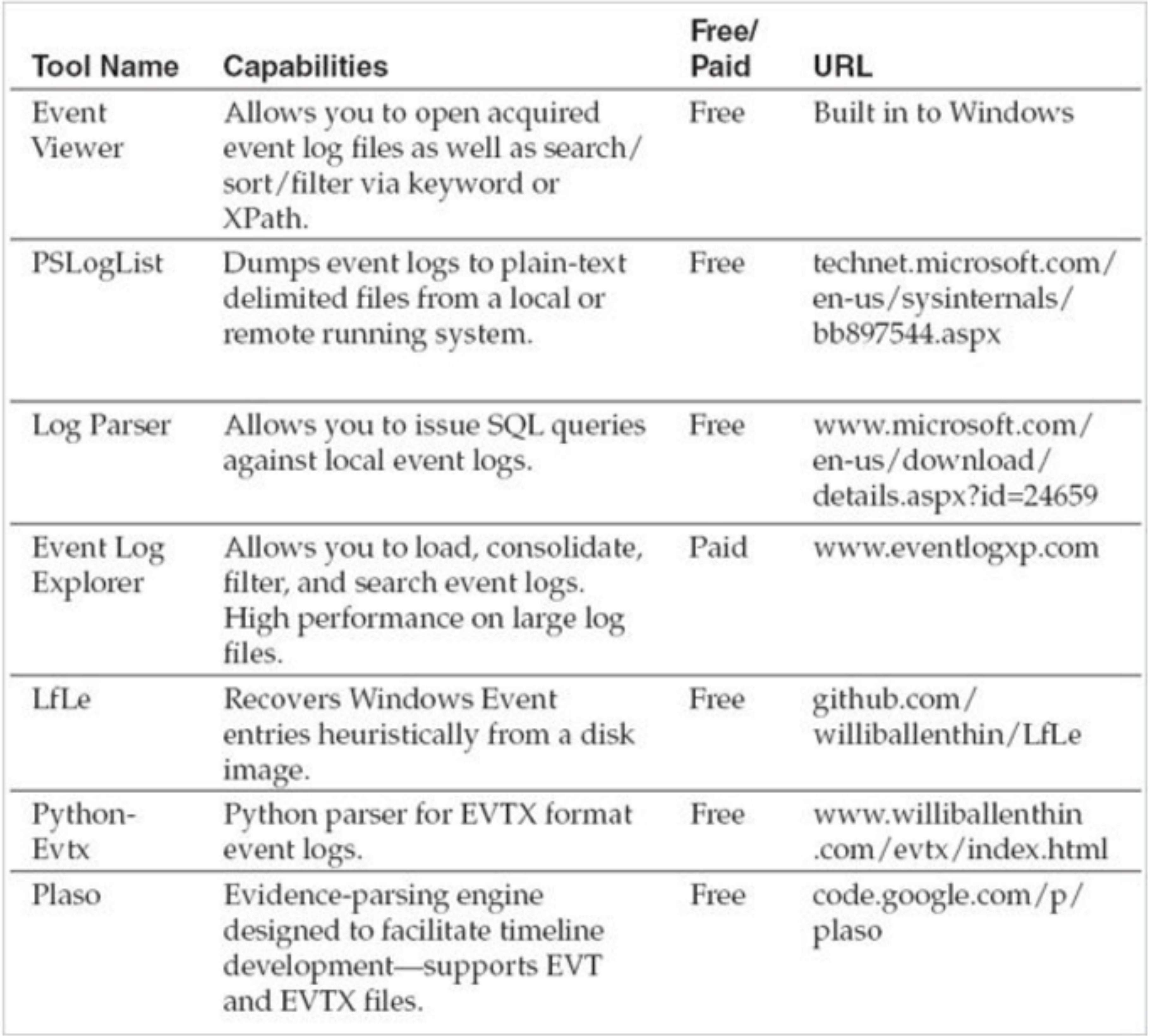
Suspicious Things

- **Abnormal usernames using PsExec**
- **Known-bad service names**
- **Errors from malicious binaries that were deleted, but still referenced by a service**

Log Analysis Tips

- **Check Application log for AV alert during period of interest**
- **Increase log file sizes to retain a longer history**
- **If log files in the old binary format are corrupt, use FixEVT (link Ch 12m)**

Tools

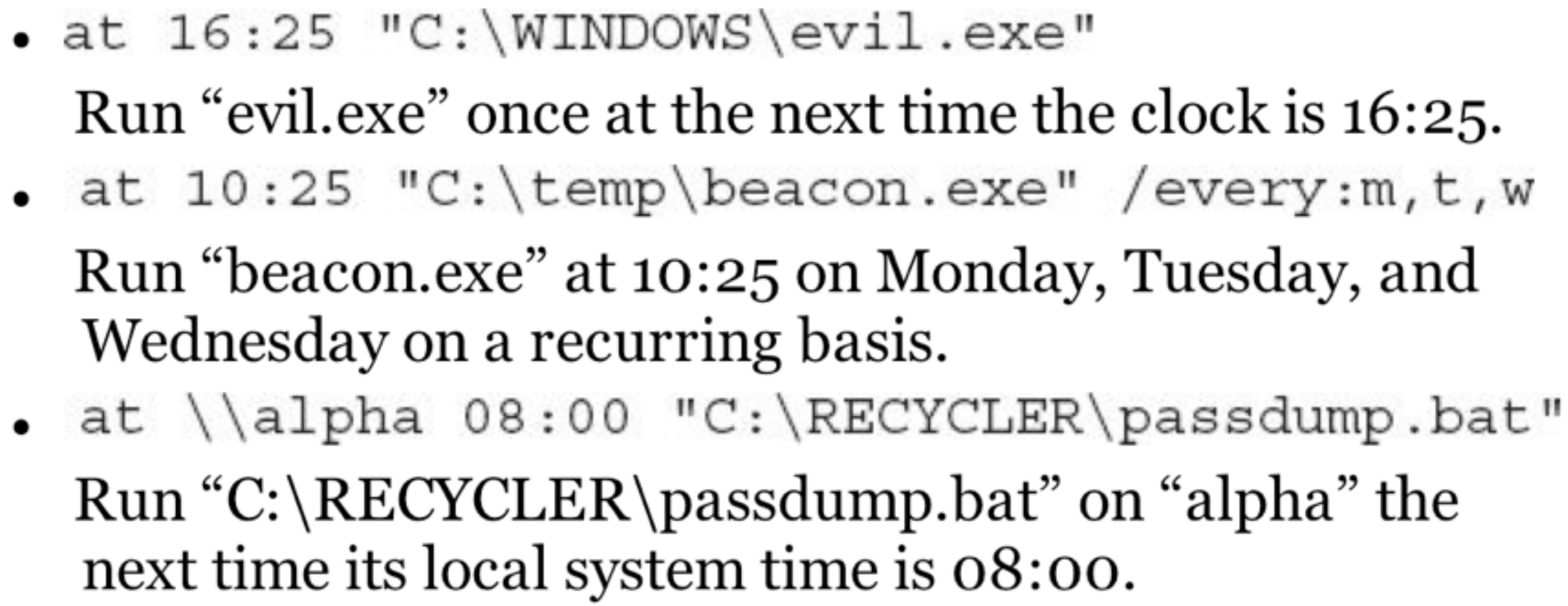


Tool Name	Capabilities	Free/ Paid	URL
Event Viewer	Allows you to open acquired event log files as well as search/ sort/filter via keyword or XPath.	Free	Built in to Windows
PSLogList	Dumps event logs to plain-text delimited files from a local or remote running system.	Free	technet.microsoft.com/ en-us/sysinternals/ bb897544.aspx
Log Parser	Allows you to issue SQL queries against local event logs.	Free	www.microsoft.com/ en-us/download/ details.aspx?id=24659
Event Log Explorer	Allows you to load, consolidate, filter, and search event logs. High performance on large log files.	Paid	www.eventlogxp.com
LfLe	Recovers Windows Event entries heuristically from a disk image.	Free	github.com/ williballenthin/LfLe
Python-Evtx	Python parser for EVTX format event logs.	Free	www.williballenthin .com/evtx/index.html
Plaso	Evidence-parsing engine designed to facilitate timeline development—supports EVT and EVTX files.	Free	code.google.com/p/ plaso

Scheduled Tasks

The "at" Command

- **Requires administrator privileges**
- **Uses local time**
- **Run as SYSTEM**



- `at 16:25 "C:\WINDOWS\evil.exe"`

  Run “evil.exe” once at the next time the clock is 16:25.
- `at 10:25 "C:\temp\beacon.exe" /every:m,t,w`

  Run “beacon.exe” at 10:25 on Monday, Tuesday, and Wednesday on a recurring basis.
- `at \\alpha 08:00 "C:\RECYCLER\passdump.bat"`

  Run “C:\RECYCLER\passdump.bat” on “alpha” the next time its local system time is 08:00.

The "schtasks" Command

- **More complex format**
- **Rarely used by attackers**

.job Files

- **Configuration data for scheduled tasks**
- **One file per task**
- **In %SYSTEMROOT%\Tasks\**
- **Files persist until shutdown or reboot of system**

Task Scheduler Logs

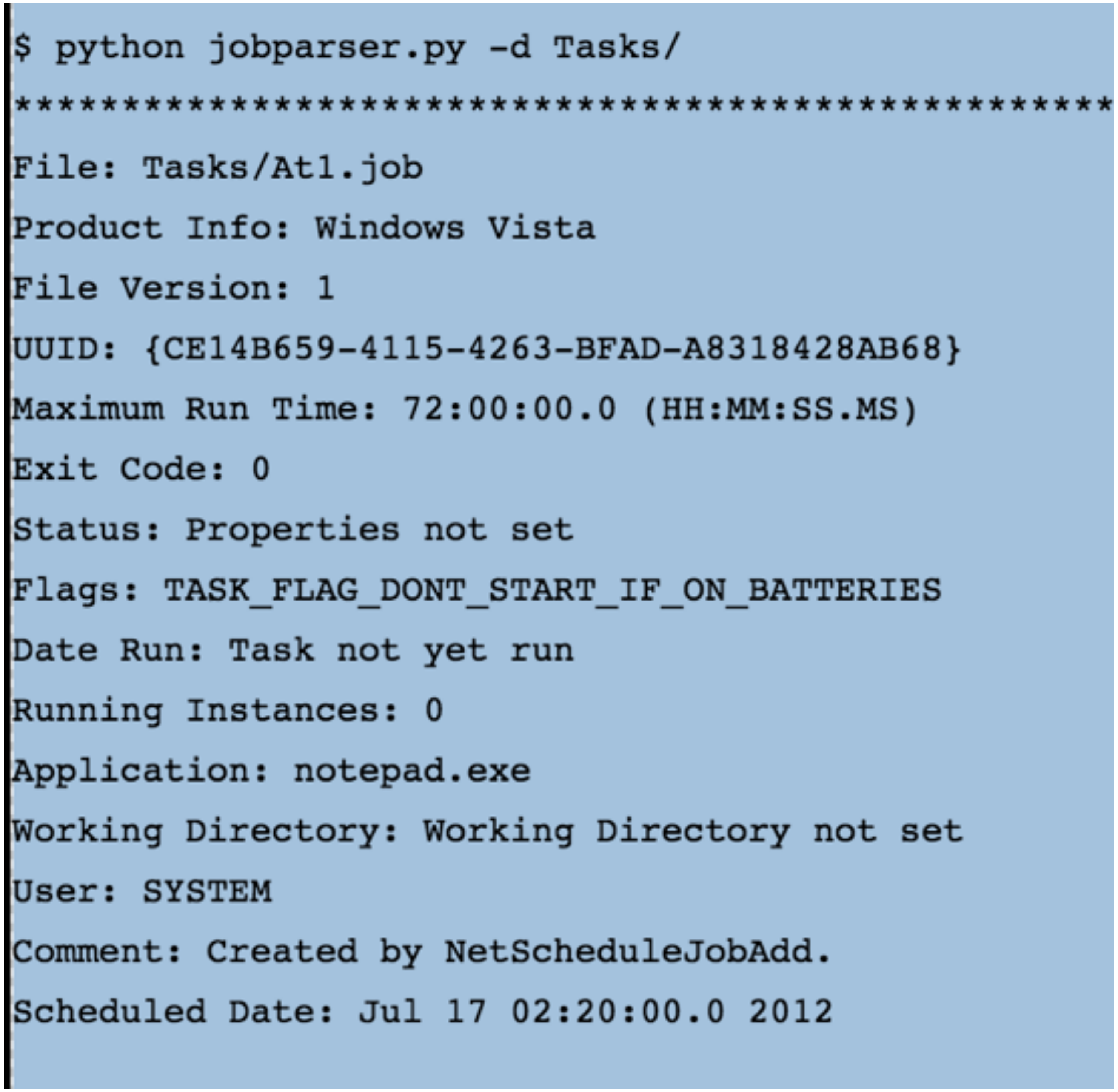
- **%SYSTEMROOT%\Tasks\SchedLgU.txt**
- **Records start time and completion of tasks**
- **Also Event Logs, including**
  - **Microsoft-Windows-TaskScheduler%4Operational.evtx**
  - **Security log**

Analyzing .job Files

- **A binary file**
- **Strings will show user information and file path**

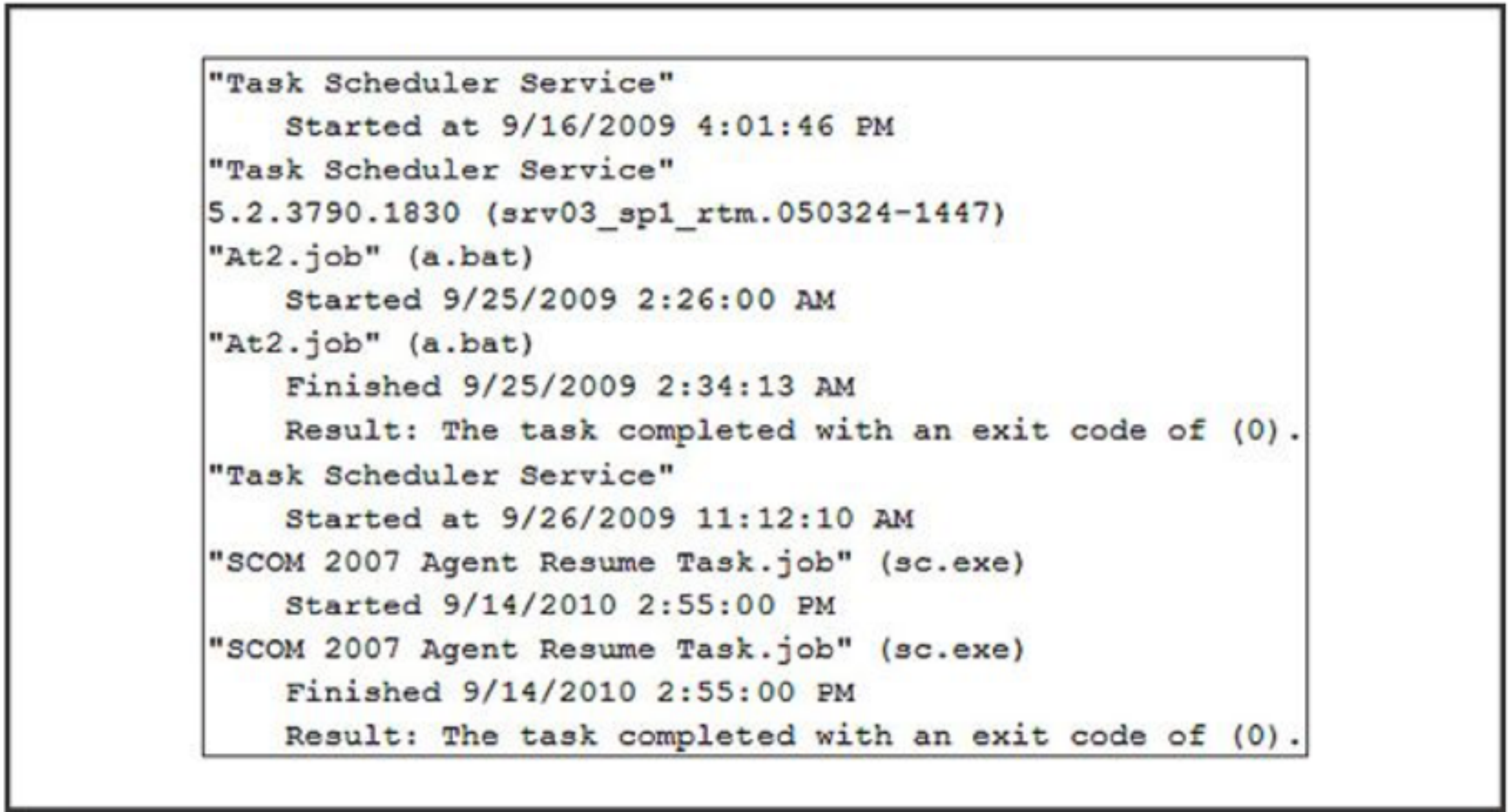
Job File Parser

- **Link Ch 12n**



```
$ python jobparser.py -d Tasks/
***************************************************
File: Tasks/At1.job
Product Info: Windows Vista
File Version: 1
UUID: {CE14B659-4115-4263-BFAD-A8318428AB68}
Maximum Run Time: 72:00:00.0 (HH:MM:SS.MS)
Exit Code: 0
Status: Properties not set
Flags: TASK_FLAG_DONT_START_IF_ON_BATTERIES
Date Run: Task not yet run
Running Instances: 0
Application: notepad.exe
Working Directory: Working Directory not set
User: SYSTEM
Comment: Created by NetScheduleJobAdd.
Scheduled Date: Jul 17 02:20:00.0 2012
```

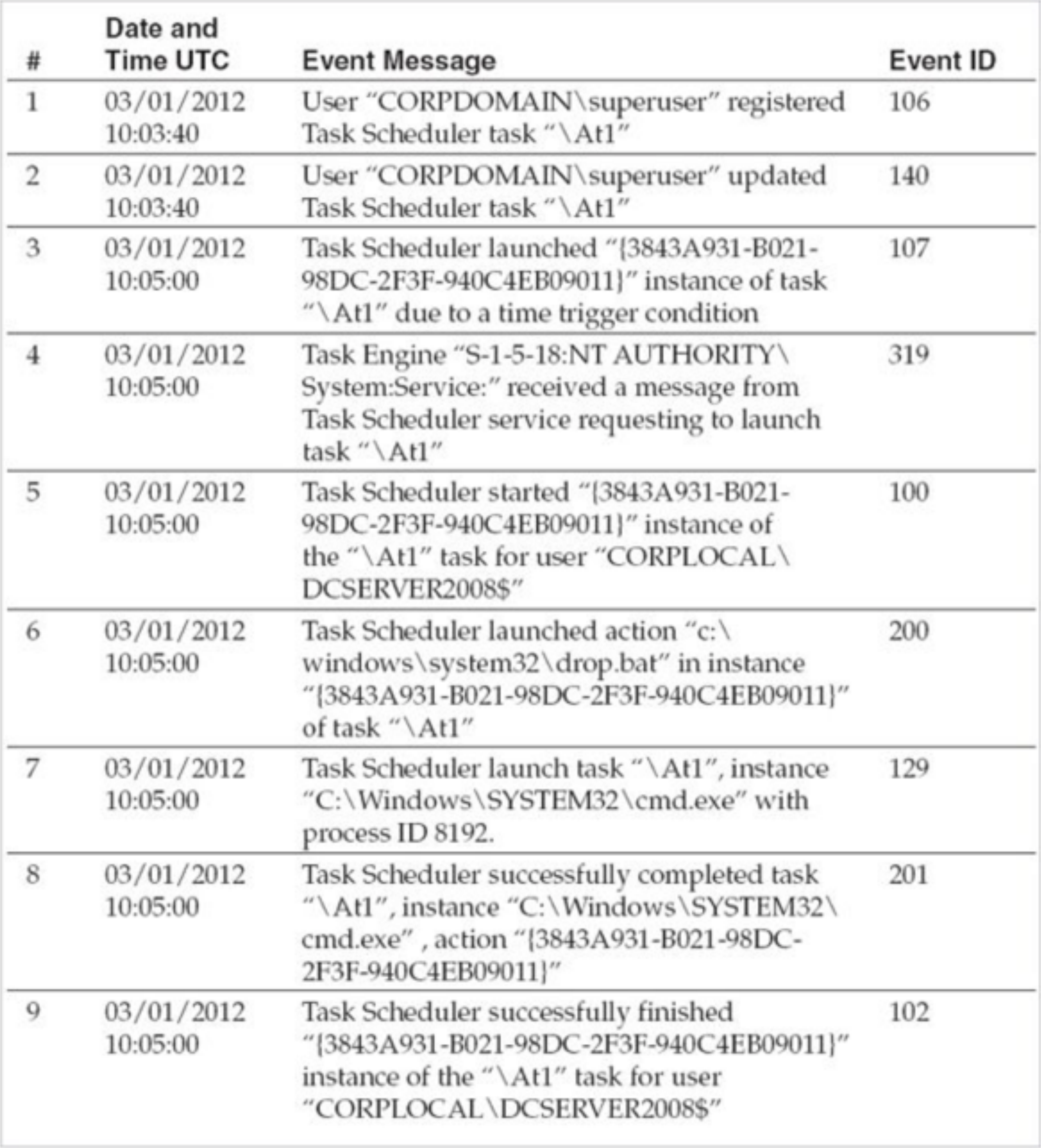
Scheduled Tasks Log



```
"Task Scheduler Service"
    Started at 9/16/2009 4:01:46 PM
"Task Scheduler Service"
5.2.3790.1830 (srv03_sp1_rtm.050324-1447)
"At2.job" (a.bat)
    Started 9/25/2009 2:26:00 AM
"At2.job" (a.bat)
    Finished 9/25/2009 2:34:13 AM
    Result: The task completed with an exit code of (0).
"Task Scheduler Service"
    Started at 9/26/2009 11:12:10 AM
"SCOM 2007 Agent Resume Task.job" (sc.exe)
    Started 9/14/2010 2:55:00 PM
"SCOM 2007 Agent Resume Task.job" (sc.exe)
    Finished 9/14/2010 2:55:00 PM
    Result: The task completed with an exit code of (0).
```

Figure 12-13. Excerpt of a sample Scheduled Task log, SchedLgU.txt

Windows Task Scheduler Operational Log in Event Viewer



#	Date and Time UTC	Event Message	Event ID
1	03/01/2012 10:03:40	User “CORPDOMAIN\superuser” registered Task Scheduler task “\At1”	106
2	03/01/2012 10:03:40	User “CORPDOMAIN\superuser” updated Task Scheduler task “\At1”	140
3	03/01/2012 10:05:00	Task Scheduler launched “{3843A931-B021-98DC-2F3F-940C4EB09011}” instance of task “\At1” due to a time trigger condition	107
4	03/01/2012 10:05:00	Task Engine “S-1-5-18:NT AUTHORITY\System:Service:” received a message from Task Scheduler service requesting to launch task “\At1”	319
5	03/01/2012 10:05:00	Task Scheduler started “{3843A931-B021-98DC-2F3F-940C4EB09011}” instance of the “\At1” task for user “CORPLOCAL\DCSERVER2008$”	100
6	03/01/2012 10:05:00	Task Scheduler launched action “c:\windows\system32\drop.bat” in instance “{3843A931-B021-98DC-2F3F-940C4EB09011}” of task “\At1”	200
7	03/01/2012 10:05:00	Task Scheduler launch task “\At1”, instance “C:\Windows\SYSTEM32\cmd.exe” with process ID 8192.	129
8	03/01/2012 10:05:00	Task Scheduler successfully completed task “\At1”, instance “C:\Windows\SYSTEM32\cmd.exe” , action “{3843A931-B021-98DC-2F3F-940C4EB09011}”	201
9	03/01/2012 10:05:00	Task Scheduler successfully finished “{3843A931-B021-98DC-2F3F-940C4EB09011}” instance of the “\At1” task for user “CORPLOCAL\DCSERVER2008$”	102