

Purple Teaming 301 - Free Attack Simulation and Alarm/Control Validation via Atomic Red

Jason Wright

The Why

The background is a solid dark blue color. On the right side, there are several overlapping, organic, leaf-like shapes in a lighter shade of blue. These shapes are curved and layered, creating a sense of depth and movement. The overall aesthetic is clean and modern.

Why are we here today?

- Have you ever wanted to verify your MSSP is properly monitoring your environment?
- Have you wanted a more cost effective way of validating security tooling and controls outside of a traditional penetration test?
- Have you ever had a pentest where our MSSP never alerted to a single command ran by the bad guys before they got domain admin?
- Have you wanted to mature your detection and response program internally through exercises without needing to hire a red team?

Intro

The background features a solid teal color with several large, overlapping, organic shapes in a darker blue shade. These shapes are positioned primarily on the right side of the frame, creating a sense of depth and movement.

Whoami

Jason Wright

- **Education**

- Masters/Bachelors from UMGC
- Assoc. from Chesapeake Community College

- **Experience**

- 4 years of technical security experience in the financial sector
- Currently a Senior Security Engineer @ Convera
- 7 years of security experience across my career
- Over a decade of experience in the information technology space
- Adjunct Faculty at Chesapeake Community College

- **Certifications**

- CISSP, GIAC GCIH, LogRhythm Security Analyst, CompTIA A+, Net+, Sec+

- **Intelligence Affiliations**

- FBI Infragard, CISA, DHS, FS-ISAC, MS-ISAC

Agenda

The background is a solid dark blue color. On the right side, there are several overlapping, curved, light blue shapes that resemble stylized leaves or petals, creating a sense of depth and movement.

Agenda - Purple Teaming 30 1

- **Already covered 'The Why' and Intros**
- **Blue, Purple, Red**
- **Technical PreReqs**
- **Lab Architecture**
- **Atomic Red T1136.001- Create a Local Account and T1136.002 Create a Domain Account**
- **Atomic Red T1055.012 - Process Injection: Process Hollowing**
- **Atomic Red T1555 – Obtaining Credentials from Password Stores**
- **How to improve**
- **Questions**
- **References**

Blue, Purple, Red

A Purple Teamers Methodology
to developing a security
program

Blue, Purple, Red

- **Blue Teaming**

- The group responsible for defending an enterprise's use of information systems by maintaining its security posture and defensive stack (NIST, n.d.). Generally, done against red teamers or adversaries.

- **Red Teaming**

- A group of individuals authorized and organized to emulate a potential adversary's attack and exploitation capabilities against an enterprise network (NIST, n.d., Red Team).

- **Purple Teaming**

- A purple team is a group of cyber professionals who simulate malicious attacks and penetration testing in order to identify security vulnerabilities and recommend strategies for mitigation (CrowdStrike, 2023).
- Unlike traditional red teams, purple teams work together closely to share information and insights to address weaknesses and improve the overall posture.

Blue, Purple, Red

- **Blue, Purple, Red**

- Blue Team – Sets up our defenses, creates alarms, monitors.
 - MSSPs, SOC, SecEng/SecOps.
 - Typically in house, affordable.
- Red Team – Attacks the defenses, finds holes and test vulnerabilities.
 - Offensive Engineers, Penetration Testers, Etc.
 - Expensive, generally contractors or vendors performing penetration tests for audit.
- Purple Team – Generally have interests and knowledge in both sides of the field.
 - Affordable, generally a blue team member that wants to explore red team side, improve defenses through continuous attack simulation versus point in time.



Technical PreRequisites

Technical PreReqs

- **Atomic Red**

- Atomic Red Team™ is a library of simple tests that every security team can execute to test their controls. Tests are focused, have few dependencies, and are defined in a structured format that can be used by automation frameworks.

- **Splunk Enterprise (SIEM) – Free Edition**

- Latest version will suffice for the experiment (great for students/home labs)
- For Enterprise – Use your own SIEM.
- WMI Must be configured for Splunk Enterprise

- **Windows Active Directory**

- Should mirror your current AD Forest and Domain Levels.
- Group policy should be copied over

- **PowerShell**

- Need to have a general understanding of powershell. Don't need to be a scripting genius, but should be able to understand what you're running and why you're running it.
- Powershell Auditing should be on for your domain. Including module and block logging.

- **Microsoft Office**

- Generally this is required for some of the scripts that try and leverage vulnerabilities/macros within Office products.

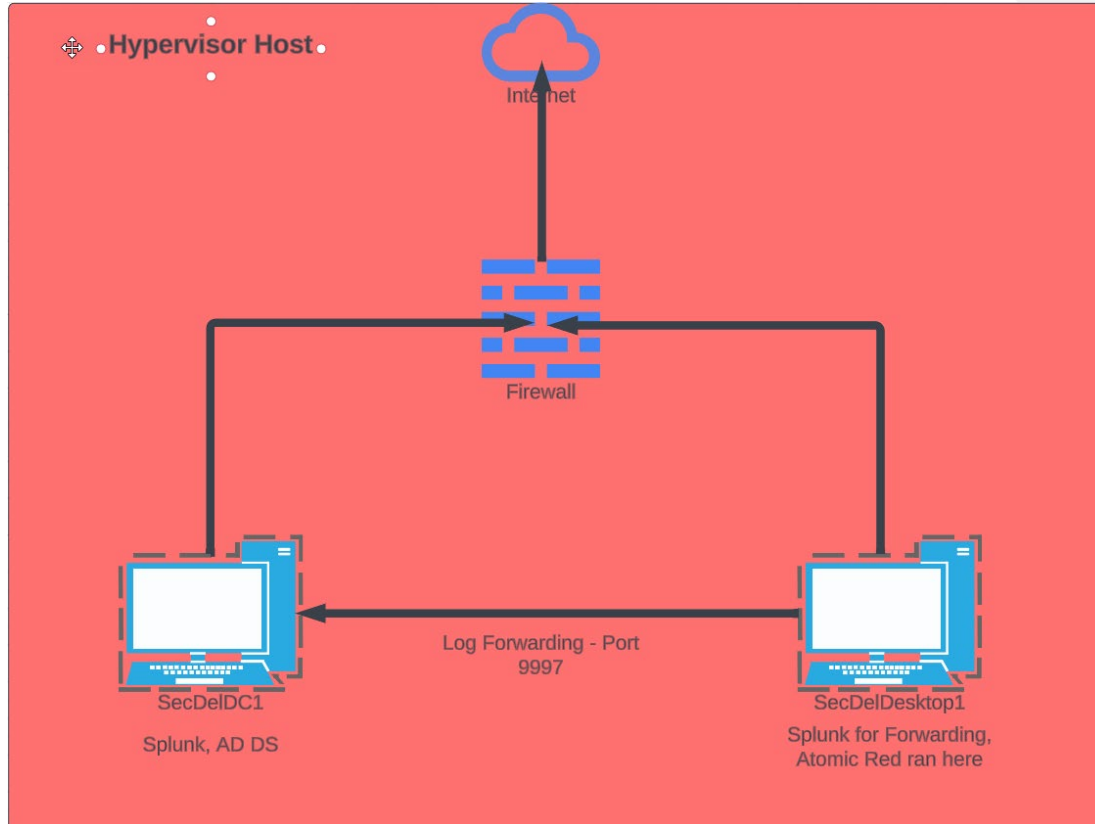
Lab Architecture

The right side of the slide features several overlapping, curved, abstract shapes in various shades of blue, ranging from a deep navy to a lighter sky blue. These shapes are layered and curved, creating a sense of depth and movement, resembling stylized architectural elements or perhaps a modern logo.

Lab Architecture

- Windows Server 2016 with AD DS Installed
- 2016 Forest and Domain Functional Levels
- Default Domain Policy – Module and Block Level Logging enabled.
- Computer Names – SecDelDC1 – Domain Controller. SecDelDesktop1 – Windows 10 Enterprise Domain Member
- Windows 10 RSAT Tools
- Both machines have splunk enterprise, with forwarding enabled on SecDelDesktop1 to SecDelDC1. NOTE: Generally not best practice to have a SIEM running on your DC.

Lab Architecture



Attack Sim 1

T1136.001- Create a Local Account
and T1136.002 Create a Domain
Account

T1136.001- Create a Local Account

- Creating a local account

Atomic Test #4 - Create a new user in PowerShell


Creates a new user in PowerShell. Upon execution, details about the new account will be displayed in the powershell session. To verify the new account, run "net user" in powershell or CMD and observe that there is a new user named "T1136.001_PowerShell"

Supported Platforms: windows

auto_generated_guid: bc8be0ac-475c-4fbf-9b1d-9ffd77afbde

Inputs:

Name	Description	Type	Default Value
username	Username of the user to create	string	T1136.001_PowerShell

 EXPLORE ATOMIC RED TEAM LEARN MORE

T1136.001

On this page

- Create Account: Local Account**
- Description from ATT&CK
- Atomic Tests
 - Atomic Test #1 - Create a user account on a Linux system
 - Atomic Test #2 - Create a user account on a MacOS system
 - Atomic Test #3 - Create a new user in a command prompt
 - Atomic Test #4 - Create a new user in PowerShell
 - Atomic Test #5 - Create a new user in Linux with 1 root UID and GID.
 - Atomic Test #6 - Create a new Windows admin user

T1136.001- Create a Local Account

- Running the attack

Attack Commands: Run with **powershell**! Elevation Required (e.g. root or admin)

```
1 New-LocalUser -Name "#{username}" -NoPassword
2
```

Cleanup Commands:

```
1 Remove-LocalUser -Name "#{username}" -ErrorAction Ignore
2
```

Administrator: Windows PowerShell

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

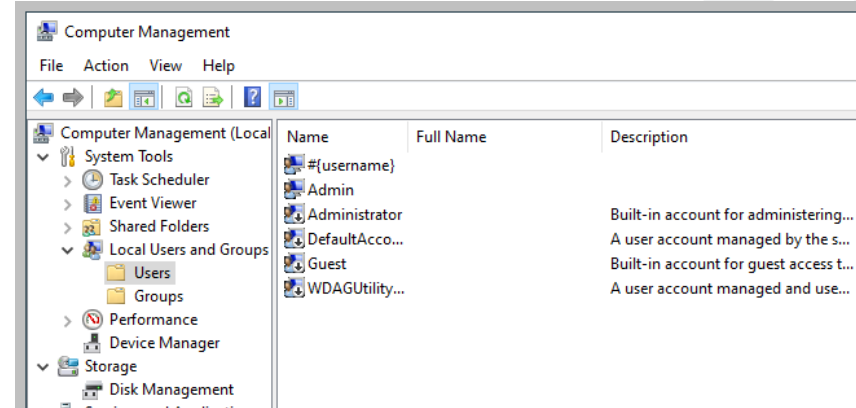
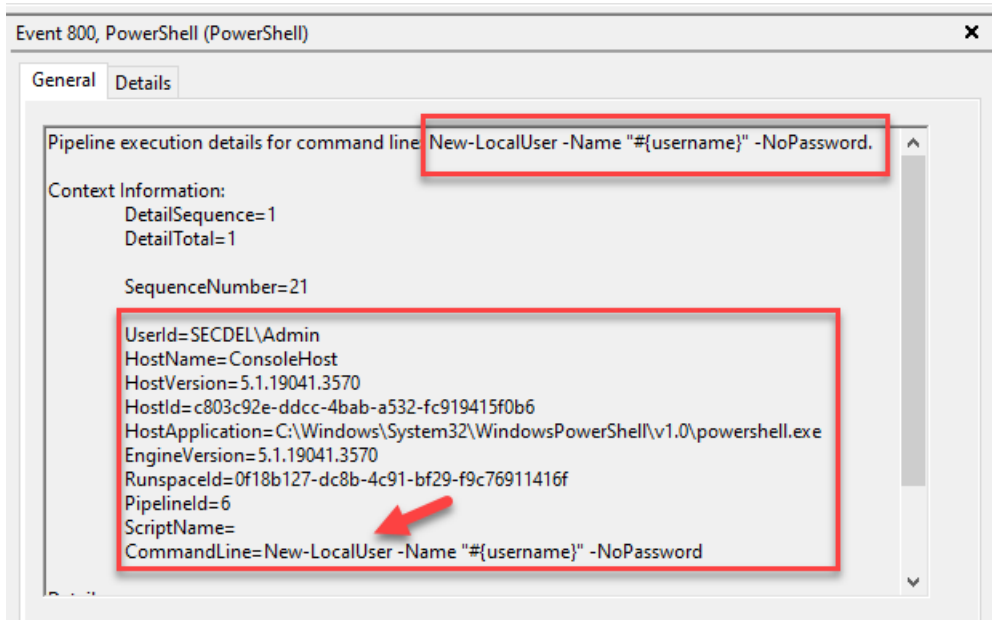
PS C:\Windows\system32> New-LocalUser -Name "#{username}" -NoPassword

Name           Enabled Description
----           -
#{username} True

PS C:\Windows\system32>
```

T1136.001- Create a Local Account

- Logging on the endpoint



T1136.001- Create a Local Account

- Logging on the SIEM
- host="secde1desktop1"
source=WinEventLog:* EventCode=800

Save As Alert

Settings

Title:

Description:

Permissions: Private Shared in App

Alert type: Scheduled Real-time

Expires:

Trigger Conditions

Trigger alert when:

Throttle?

Trigger Actions

+ Add Actions

- Add to Triggered Alerts
- Log Event
- Output results to lookup
- Output results to telemetry endpoint
- Run a script
- Send email

Cancel Save

```
> 10/15/23 10/15/2023 07:35:39 PM
7:35:39.000 PM LogName=Windows PowerShell
EventCode=800
EventType=4
ComputerName=SecDelDesktop1.SecDel1.internal
SourceName=PowerShell
Type=Information
RecordNumber=3335
Keywords=Classic
TaskCategory=Pipeline Execution Details
OpCode=Info
Message=Pipeline execution details for command line: New-LocalUser -Name "#{username}" -NoPassword.

Context Information:
DetailSequence=1
DetailTotal=1

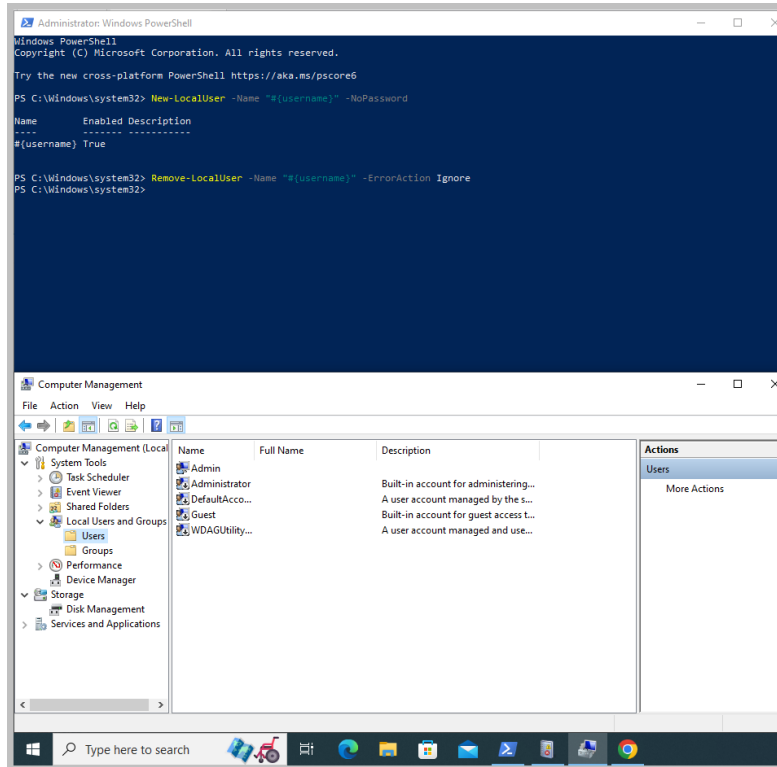
SequenceNumber=21

UserId=SECDEL\Admin
HostName=ConsoleHost
HostVersion=5.1.19041.3570
HostId=c803c92e-ddcc-4bab-a532-fc919415f0b6
HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
EngineVersion=5.1.19041.3570
RunspaceId=0f18b127-dc8b-4c91-bf29-f9c76911416f
PipelineId=6
ScriptName=
CommandLine=New-LocalUser -Name "#{username}" -NoPassword

Details:
CommandInvocation(New-LocalUser): "New-LocalUser"
ParameterBinding(New-LocalUser): name="Name"; value="#{username}"
ParameterBinding(New-LocalUser): name="NoPassword"; value="True"
Collapse
host = SECDELDESKTOP1 | source = WinEventLog:Windows PowerShell
sourcetype = WinEventLog:Windows PowerShell
```

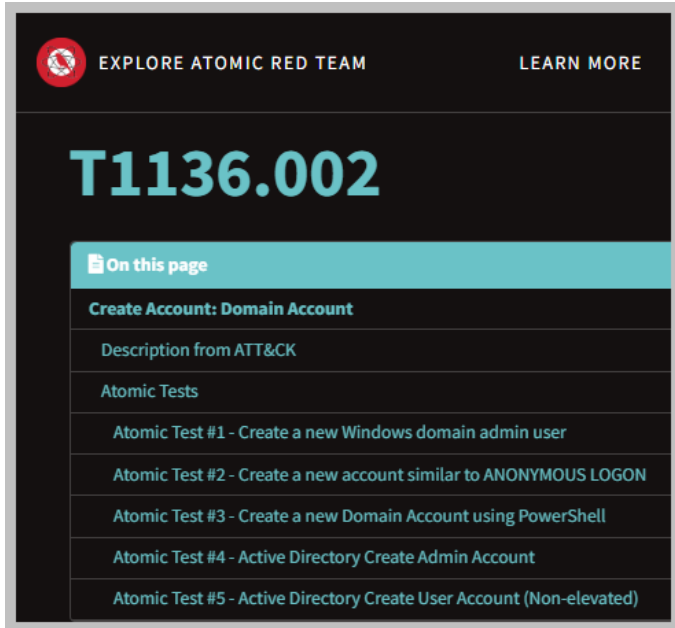
T1136.001- Create a Local Account

- Clean Up



T1136.002 Create a Domain Account

- Creating a domain account



EXPLORE ATOMIC RED TEAM [LEARN MORE](#)

T1136.002

On this page

- Create Account: Domain Account**
- Description from ATT&CK
- Atomic Tests
 - Atomic Test #1 - Create a new Windows domain admin user
 - Atomic Test #2 - Create a new account similar to ANONYMOUS LOGON
 - Atomic Test #3 - Create a new Domain Account using PowerShell
 - Atomic Test #4 - Active Directory Create Admin Account
 - Atomic Test #5 - Active Directory Create User Account (Non-elevated)

Name	Description	Type	Default Value
username	Name of the Account to be created	string	T1136.002_Admin
password	Password of the Account to be created	string	T1136_pass123!

Attack Commands: Run with **powershell!**

```
1 $SamAccountName = '#{username}'
2 $AccountPassword = ConvertTo-SecureString '#{password}' -AsPla:
3 Add-Type -AssemblyName System.DirectoryServices.AccountManagem
4 $Context = New-Object -TypeName System.DirectoryServices.Accou
5 $User = New-Object -TypeName System.DirectoryServices.AccountM
6 $User.SamAccountName = $SamAccountName
7 $TempCred = New-Object System.Management.Automation.PSCredenti
8 $User.SetPassword($TempCred.GetNetworkCredential().Password)
9 $User.Enabled = $True
10 $User.PasswordNotRequired = $False
11 $User.DisplayName = $SamAccountName
12 $User.Save()
13 $User
14
```

Cleanup Commands:

```
1 cmd /c "net user #{username} /del >nul 2>&1"
2
```

T1136.002 Create a Domain Account

- Running the attack

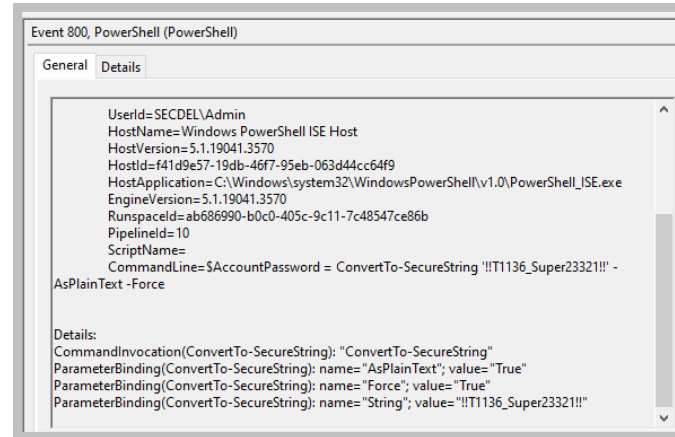
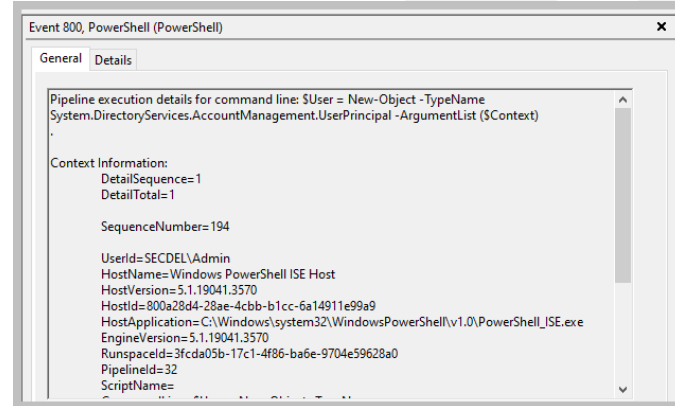
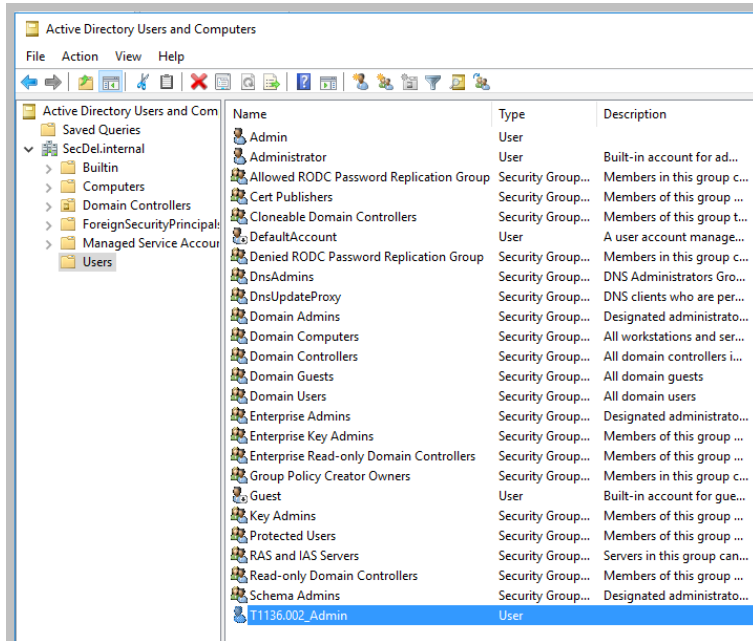
```
1 $SamAccountName = 'T1136.002_Admin'
2 $AccountPassword = ConvertTo-SecureString 'T1136_pass123!' -AsPlainText -Force
3 Add-Type -AssemblyName System.DirectoryServices.AccountManagement
4 $Context = New-Object -TypeName System.DirectoryServices.AccountManagement.PrincipalContext -A
5 $User = New-Object -TypeName System.DirectoryServices.AccountManagement.UserPrincipal -A
6 $User.SamAccountName = $SamAccountName
7 $TempCred = New-Object System.Management.Automation.PSCredential('a', $AccountPassword)
8 $User.SetPassword($TempCred.GetNetworkCredential().Password)
9 $User.Enabled = $True
10 $User.PasswordNotRequired = $False
11 $User.DisplayName = $SamAccountName
12 $User.Save()
13 $User
14
```

```
Exception calling 'Save' with '0' argument(s): 'The password does not meet the password policy req
length, password complexity and password history requirements. (Exception from HRESULT: 0x800708C5)
At line:12 char:1
+ $User.Save()
+ ~~~~~
+ CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
+ FullyQualifiedErrorId : PasswordException

GivenName                :
MiddleName               :
Surname                  :
EmailAddress              :
VoiceTelephoneNumber     :
EmployeeId               :
AdvancedSearchFilter     : System.DirectoryServices.AccountManagement.AdvancedFilters
Enabled                  : True
AccountLockoutTime       :
LastLogon                :
PermittedWorkstations    : {}
PermittedLogonTimes      :
AccountExpirationDate    :
SmartcardLogonRequired  : False
DelegationPermitted     : False
BadLogonCount            : 0
HomeDirectory            :
HomeDrive                :
ScriptPath               :
LastPasswordSet          :
LastBadPasswordAttempt   :
PasswordNotRequired     : False
PasswordNeverExpires    : False
UserCannotChangePassword : False
AllowReversiblePasswordEncryption : False
Certificates              : {}
Context                  : System.DirectoryServices.AccountManagement.PrincipalContext
ContextType              : Domain
Description              :
DisplayName               : T1136.002_Admin
SamAccountName           : T1136.002_Admin
UserPrincipalName        :
Sid                      :
Guid                     :
DistinguishedName        :
StructuralObjectClass    :
Name                     :
```

T1136.002 Create a Domain Account

- Logging on the endpoint



T1136.002 Create a Domain Account

- Logging on the SIEM

i	Time	Event
>	10/15/23 8:18:30.000 PM	<pre>10/15/2023 08:18:30 PM LogName=Windows PowerShell EventCode=800 EventType=4 ComputerName=SecDelDesktop1.SecDel.internal SourceName=PowerShell Type=Information RecordNumber=3462 Keywords=Classic TaskCategory=Pipeline Execution Details OpCode=Info Message=Pipeline execution details for command line: \$AccountPassword = ConvertTo-SecureString '!!T1136_Super2332 !!!' -AsPlainText -Force . Context Information: DetailSequence=1 DetailTotal=1 SequenceNumber=35 UserId=SECDEL\Admin HostName=Windows PowerShell ISE Host HostVersion=5.1.19041.3570 HostId=f41d9e57-19db-46f7-95eb-963d44cc64f9 HostApplication=C:\Windows\system32\WindowsPowerShell\v1.0\PowerShell_ISE.exe EngineVersion=5.1.19041.3570 RunspaceId=ab686990-b8c0-405c-9c11-7c48547ce86b PipelineId=10 ScriptName= CommandLine=\$AccountPassword = ConvertTo-SecureString '!!T1136_Super2332!!!' -AsPlainText -Force Details: CommandInvocation(ConvertTo-SecureString): "ConvertTo-SecureString" ParameterBinding(ConvertTo-SecureString): name="AsPlainText"; value="True" ParameterBinding(ConvertTo-SecureString): name="Force"; value="True" ParameterBinding(ConvertTo-SecureString): name="String"; value="!!T1136_Super2332!!!" Collapse host = SECDELDESKTOP1 source = WinEventLog:Windows PowerShell sourcetype = WinEventLog:Windows PowerShell</pre>

Reflecting

- T1136.001- Create a Local Account and T1136.002 Create a Domain Account
- Do we have the proper logging on the endpoint?
- Do we have the proper logging on the SIEM?
- Are we being alerted?
- What don't we care about?
- Are we sending the logs in a format that can be parsed?

Attack Sim 2

Atomic Red T1055.012 - Process
Injection: Process Hollowing

Atomic Red T10 55.0 12 - Process Injection

- What is Process Injection
- What is Process Hollowing

Atomic Red T10 55.0 12 - Process Injection

- Showing of how to run the attack.

Atomic Test #1 - Process Hollowing using PowerShell

This test uses PowerShell to create a Hollow from a PE on disk with explorer as the parent. Credit to FuzzySecurity (<https://github.com/FuzzySecurity/PowerShell-Suite/blob/master/Start-Hollow.ps1>)

Supported Platforms: windows

auto_generated_guid: 562427b4-39ef-4e8c-af88-463a78e70b9c

Inputs:

Name	Description	Type	Default Value
hollow_binary_path	Path of the binary to hollow (executable that will run inside the sponsor)	string	C:\Windows\System32\cmd.exe
parent_process_name	Name of the parent process	string	explorer
sponsor_binary_path	Path of the sponsor binary (executable that will host the binary)	string	C:\Windows\System32\notepad.exe
spawnto_process_name	Name of the process to spawn	string	notepad

Attack Commands: Run with powershell!

```
1 | . "$PathToAtomicsFolder\T1055.012\src\Start-Hollow.ps1"
2 | $ppid=Get-Process #{parent_process_name} | select -expand id
3 | Start-Hollow -Sponsor "#{sponsor_binary_path}" -Hollow "#{hollow_binary_path}"
4 |
```

Cleanup Commands:

```
1 | Stop-Process -Name "#{spawnto_process_name}" -ErrorAction Ignore
2 |
```

Atomic Red T10 55.0 12 - Process Injection

- Running the Attack

The screenshot displays two windows from a Windows system. The primary window is the Windows PowerShell ISE, titled 'Administrator: Windows PowerShell ISE'. It shows a script named 'Start-Hollow.ps1' with the following commands:

```
1 "C:\Users\admin.SECDEL\Documents\Start-Hollow.ps1"
2 $ppid=Get-Process explorer | select -expand id
3 Start-Hollow -Sponsor "C:\Windows\System32\notepad.exe" -Hollow "C:\Windows\System32\cmd.exe" -ParentPID $ppid
```

Below the script, the execution output is visible, showing detailed verbose messages from the 'Start-Hollow' function, such as 'Created section from file handle', 'Acquired PEI', and 'Created hollow process parameters'. The prompt shows the command was executed successfully, returning 'True'.

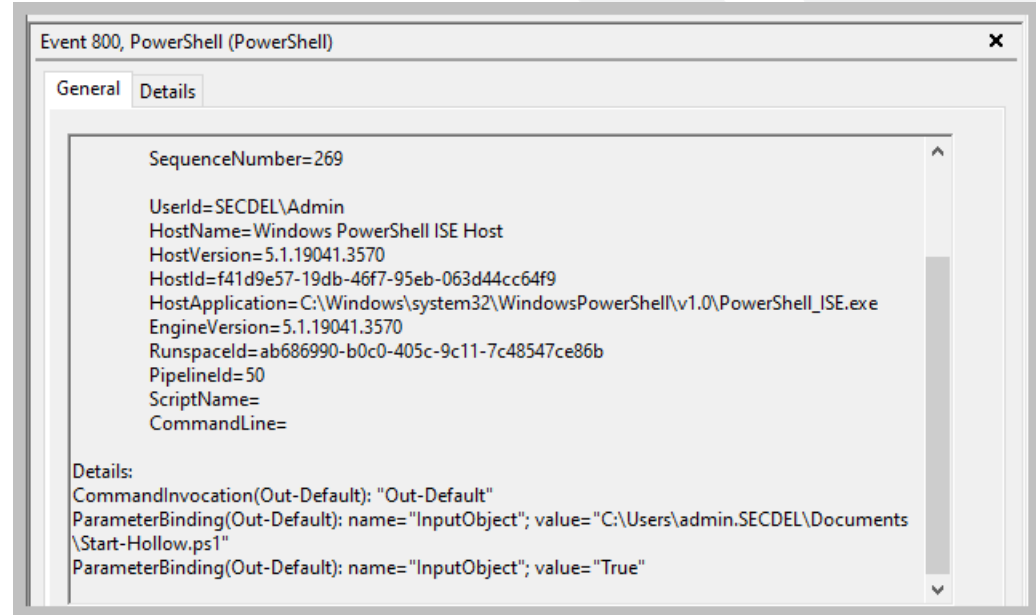
The secondary window is the Event Viewer, titled 'Event Viewer'. It shows a log entry for 'Hollow' with the following details:

- Level: Information
- Source: Windows PowerShell
- Number of events: 3,471
- Message: (c) Microsoft Corporation. All rights reserved. Not enough memory resources are available to process this command. C:\Windows\System32>

Red text labels are overlaid on the image: 'Scripts in ISE' points to the PowerShell script, and 'Hollowed Session' points to the Event Viewer log entry.

Atomic Red T10 55.0 12 - Process Injection

- Logging on the Endpoint
- Always check with your EDR when it comes to processes!



Atomic Red T10 55.0 12 - Process Injection

- Showing of logging on the SIEM.

```
Event
10/15/2023 08:58:21 PM
LogName=Windows PowerShell
EventCode=800
EventType=4
ComputerName=SecDelDesktop1.SecDel.internal
SourceName=PowerShell
Type=Information
RecordNumber=3578
Keywords=Classic
TaskCategory=Pipeline Execution Details
OpCode=Info
Message=Pipeline execution details for command line: .

Context Information:
  DetailSequence=1
  DetailTotal=1

  SequenceNumber=269

  UserId=SECDEL\Admin
  HostName=Windows PowerShell ISE Host
  HostVersion=5.1.19041.3570
  HostId=f41d9e57-19db-46f7-95eb-063d44cc64f9
  HostApplication=C:\Windows\system32\WindowsPowerShell\v1.0\PowerShell_ISE.exe
  EngineVersion=5.1.19041.3570
  RunspaceId=ab686990-b0c0-405c-9c11-7c48547ce86b
  PipelineId=50
  ScriptName=
  CommandLine=

Details:
CommandInvocation(Out-Default): "Out-Default"
ParameterBinding(Out-Default): name="InputObject"; value="C:\Users\admin.SECDEL\Documents\Start-Hollow.ps1"
ParameterBinding(Out-Default): name="InputObject"; value="True"
Collapse
host = SECDELDESKTOP1 | source = WinEventLog:Windows PowerShell | sourcetype = WinEventLog:Windows PowerShell
```


Reflecting

- Atomic Red T1055.012 - Process Injection: Process Hollowing
- Do we have the proper logging on the endpoint?
- Do we have the proper logging on the SIEM?
- Are we being alerted by the EDR?
- What don't we care about?
- Are we sending the logs in a format that can be parsed?
- Should we be forwarding logs from the EDR into the SIEM? Alerts and otherwise?

Attack Sim 3

Atomic Red T1555 – Obtaining
Credentials from Password Stores

Atomic Red T1555 – Obtaining Credentials from Password Stores

- Various cached credential locations on workstations
- The SAM Hive of the Registry

Atomic Red T1555 – Obtaining Credentials from Password Stores

- How to run the attack

Atomic Test #1 - Registry dump of SAM, creds, and secrets [↗](#)

Local SAM (SAM & System), cached credentials (System & Security) and LSA secrets (System & Security) can be enumerated via three registry keys. Then processed locally using <https://github.com/Neohapsis/creddump7>

Upon successful execution of this test, you will find three files named, sam, system and security in the %temp% directory.

Supported Platforms: Windows

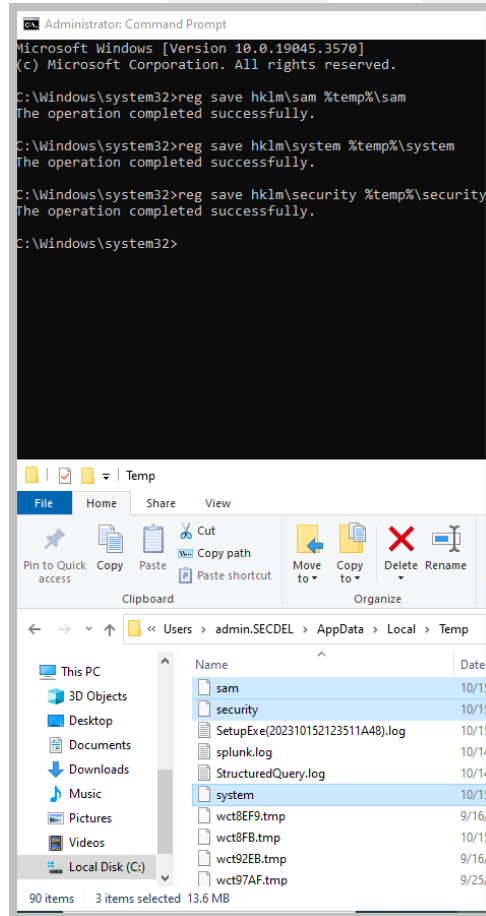
auto_generated_guid: 5c2571d0-1572-416d-9676-812e64ca9f44

Attack Commands: Run with `command_prompt`! Elevation Required (e.g. root or admin) [↗](#)

```
reg save HKLM\sam %temp%\sam
reg save HKLM\system %temp%\system
reg save HKLM\security %temp%\security
```

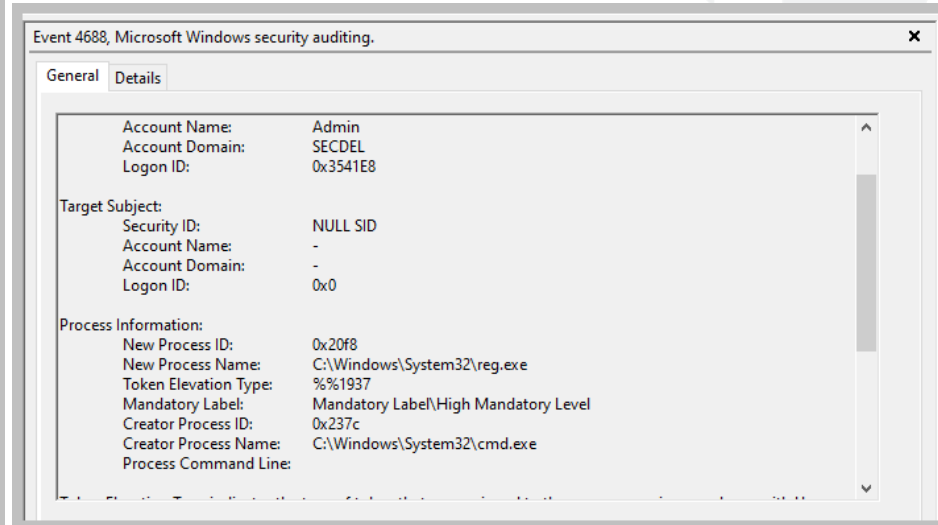
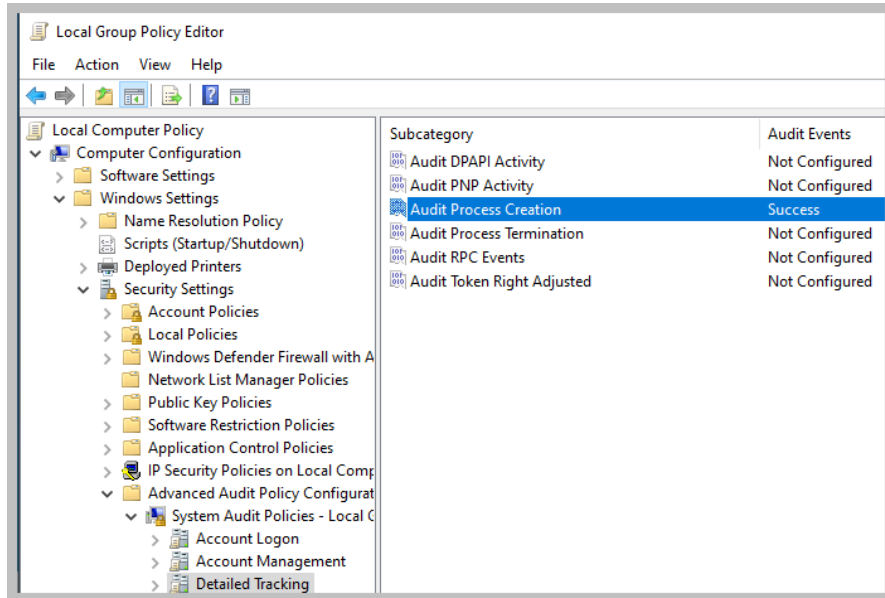
Cleanup Commands: [↗](#)

```
del %temp%\sam >nul 2> nul
del %temp%\system >nul 2> nul
del %temp%\security >nul 2> nul
```



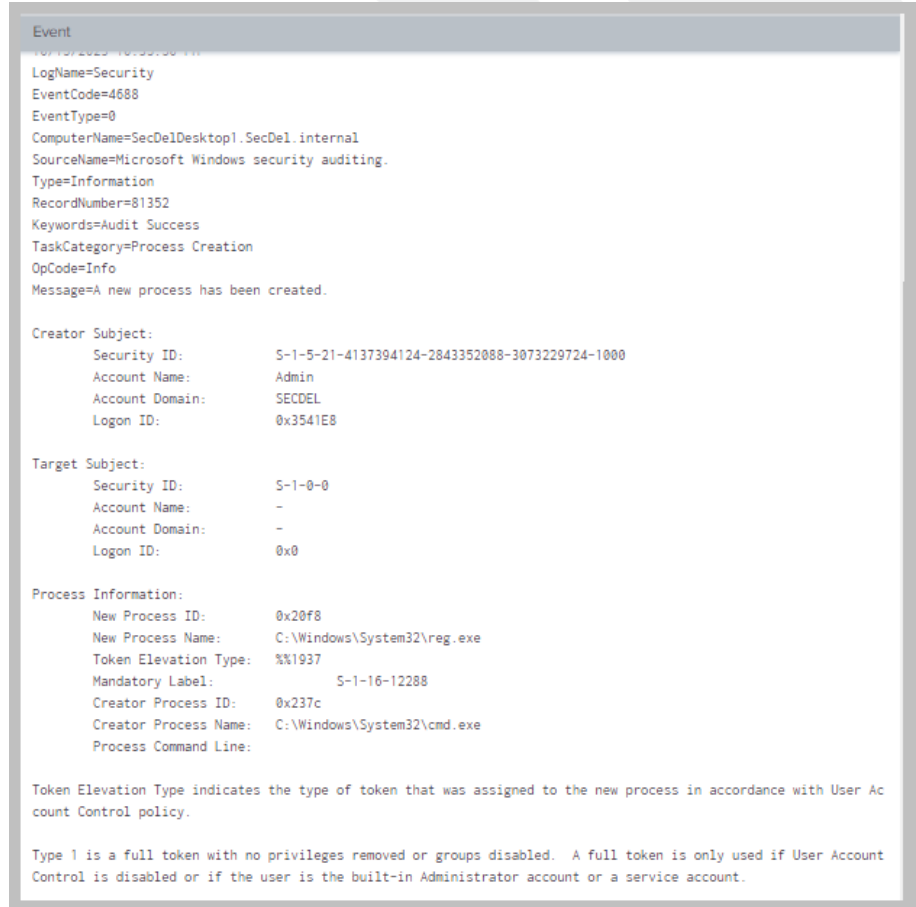
Atomic Red T1555 – Obtaining Credentials from Password Stores

- What does logging look like on the endpoint



Atomic Red T1555 – Obtaining Credentials from Password Stores

- What does logging look like on the SIEM?



The screenshot displays a Windows Security event log entry. The event is titled 'Event' and contains the following details:

- LogName=Security
- EventCode=4688
- EventType=0
- ComputerName=SecDelDesktop1.SecDel.internal
- SourceName=Microsoft Windows security auditing.
- Type=Information
- RecordNumber=81352
- Keywords=Audit Success
- TaskCategory=Process Creation
- OpCode=Info
- Message=A new process has been created.

The event details are organized into three sections:

- Creator Subject:**
 - Security ID: S-1-5-21-4137394124-2843352088-3073229724-1000
 - Account Name: Admin
 - Account Domain: SECDEL
 - Logon ID: 0x3541E8
- Target Subject:**
 - Security ID: S-1-0-0
 - Account Name: -
 - Account Domain: -
 - Logon ID: 0x0
- Process Information:**
 - New Process ID: 0x20f8
 - New Process Name: C:\Windows\System32\reg.exe
 - Token Elevation Type: %%1937
 - Mandatory Label: S-1-16-12288
 - Creator Process ID: 0x237c
 - Creator Process Name: C:\Windows\System32\cmd.exe
 - Process Command Line:

Token Elevation Type indicates the type of token that was assigned to the new process in accordance with User Account Control policy.

Type 1 is a full token with no privileges removed or groups disabled. A full token is only used if User Account Control is disabled or if the user is the built-in Administrator account or a service account.

Reflection

- Atomic Red T1555 – Obtaining Credentials from Password Stores

```
Event
10/10/2025 10:55:50 AM
LogName=Security
EventCode=4688
EventType=0
ComputerName=SecDelDesktop1.SecDel.internal
SourceName=Microsoft Windows security auditing.
Type=Information
RecordNumber=81352
Keywords=Audit Success
TaskCategory=Process Creation
OpCode=Info
Message=A new process has been created.

Creator Subject:
Security ID:          S-1-5-21-4137394124-2843352088-3073229724-1000
Account Name:        Admin
Account Domain:      SECDEL
Logon ID:             0x3541E8

Target Subject:
Security ID:          S-1-0-0
Account Name:        -
Account Domain:      -
Logon ID:             0x0

Process Information:
New Process ID:       0x20f8
New Process Name:    C:\Windows\System32\reg.exe
Token Elevation Type: %1937
Mandatory Label:     S-1-16-12288
Creator Process ID:  0x237c
Creator Process Name: C:\Windows\System32\cmd.exe
Process Command Line:

Token Elevation Type indicates the type of token that was assigned to the new process in accordance with User Account Control policy.

Type 1 is a full token with no privileges removed or groups disabled. A full token is only used if User Account Control is disabled or if the user is the built-in Administrator account or a service account.
```

A large, stylized graphic of a leaf or petal in shades of blue, positioned on the right side of the slide. It has a dark blue outline and a lighter blue fill, with several curved segments suggesting its shape.

How do we improve

Improvements

- Powershell Auditing – By default this is turned off! Be as verbose as possible!
- Script block logging – also turned off by default.
- Command Prompt Logging – Guess what?
- Logging all of the things! Domain Controllers. Endpoints. Central logging. Web applications?
- Test your MSSP. Their job is to protect you. Run simulations regularly and test their alerting capacity. If they don't alarm, work with them to create better parsers to improve not only alerting for your environment, but for their other customers as well.
- Automate Testing. Mature this out further by automating this testing and tracking the results to resolutions. This helps to prove technical KPIs that executives can understand. Free products such as Vectr, another purple teaming utility, allows you to fully automate and track the results of simulations

Improvements

- Find your crown jewels in your environment, outside of Domain Controllers or Authentication. What drives the business?
- Let the blue teamers hone their craft. An organization doesn't need a red team or annual penetration test, unless for compliance of course.
- Incorporate threat intelligence into your pre-existing alerting through STIX and TAXI feeds.

Questions

The background features a solid teal color with several large, overlapping, organic shapes in a darker blue shade on the right side, creating a layered, leaf-like or wave-like effect.

References

- NIST. (n.d.). Blue Team - NIST Glossary. NIST Glossary. https://csrc.nist.gov/glossary/term/blue_team
- NIST. (n.d.). Red Team - NIST Glossary. NIST Glossary. https://csrc.nist.gov/glossary/term/red_team
- CrowdStrike. (2023, February 24). What is a purple team?. crowdstrike.com. <https://www.crowdstrike.com/cybersecurity-101/purple-teaming/>
- <https://atomicredteam.io/>
- <https://atomicredteam.io/atomics/>
- <https://github.com/redcanaryco/atomic-red-team/wiki/>

References

- <https://atomicredteam.io/persistence/T1136.001/>
- <https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1055.012/T1055.012.md#atomic-test-1---process-hollowing-using-powershell>
- <https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1003.002/T1003.002.md#atomic-test-1---registry-dump-of-sam-creds-and-secrets>
- https://www.splunk.com/en_us/products/splunk-enterprise.html
- <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/component-updates/command-line-process-auditing>