DEFENSIVE POWERSHELL

JAMES HONEYCUTT

ABOUT ME

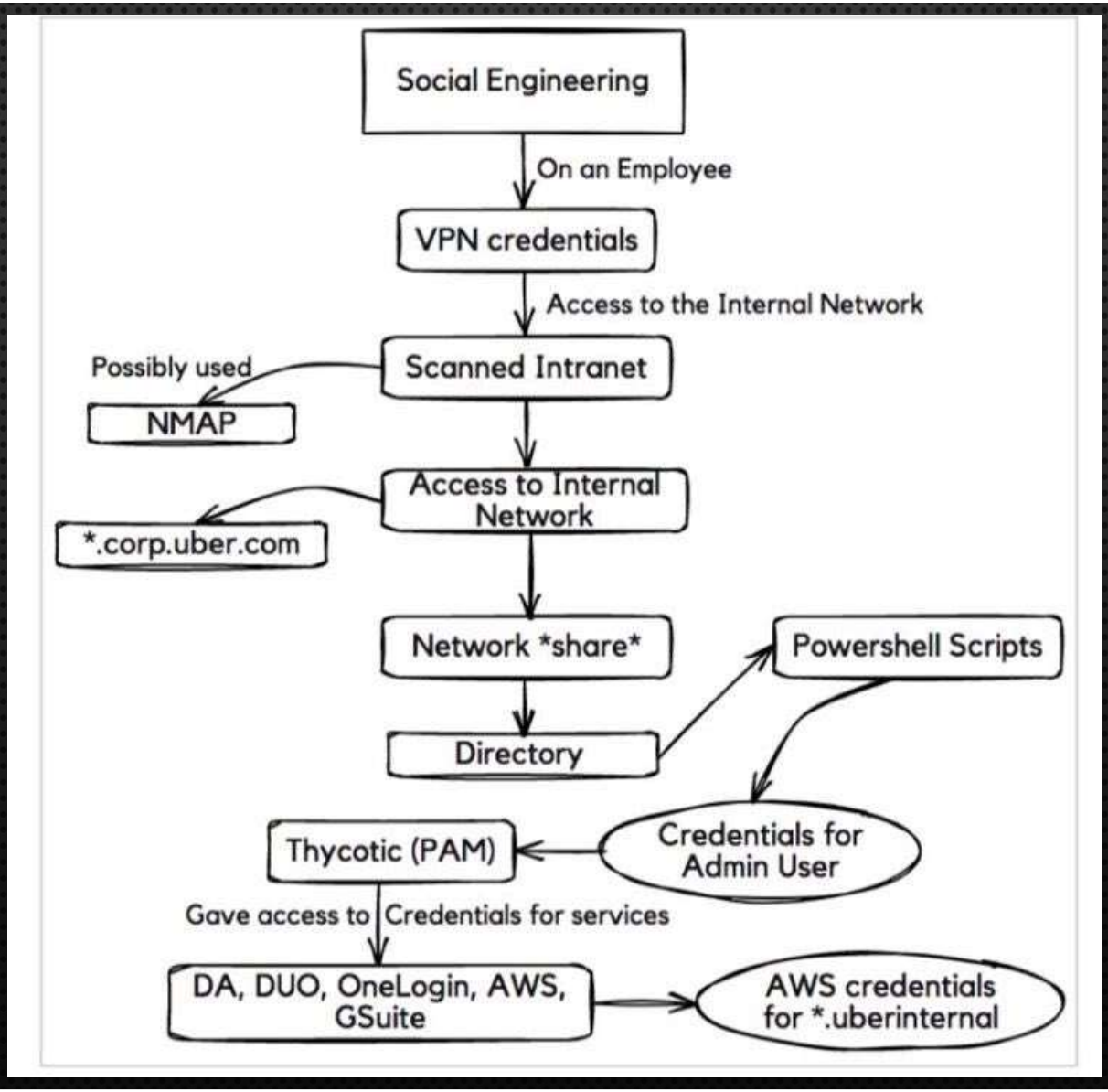
- OVER 20 YEARS IN IT/CYBER
- SELF-TAUGHT POWERSHELL BACK IN POWERSHELL V2
- POWERSHELL WAS MAIN JOB FUNCTION DURING MY LAST TWO POSITIONS

LEARNER INTRO

- NAME
- POWERSHELL EXPERIENCE
- WHAT DO YOU HOPE TO GET OUT OF THIS WORKSHOP

UBER HACK


https://docs.microsoft.com/en-us/powershell/module/defender/?view=windowsserver2022-ps
UBER


Social Engineering
On an Employee
VPN credentials
Access to the Internal Network
Possibly used
Scanned Intranet
NMAP
Access to Internal Network
*.corp.uber.com
Network *share*
Powershell Scripts
Directory
Thycotic (PAM)
Credentials for Admin User
Gave access to
Credentials for services
DA, DUO, OneLogin, AWS, GSuite
AWS credentials for *.uberinternal


ENVIRONMENT SETUP

- Download and install a Virtual Windows machine
- Download Git Repo
  - https://github.com/P0w3rChi3f/Defensive-PowerShell
- Download install VSCode (Optional)
  - https://code.visualstudio.com/download

LEARNING OBJECTIVE

- Action: Learner will identify ways to use PowerShell in a Defensive manner
- Condition: Given learning activities, readings, peer and instructor feedback, reflection time, development time, and practical exercises.
- Standard: Learner will identify a Defensive PowerShell technique to use in a given situation


OVERVIEW

- PowerShell Review
- PowerShell Remoting Review
- Securing Windows with PowerShell
- Hunting with PowerShell (Log Parsing)

POWERSHELL REVIEW

- POWERSHELL HISTORY
- THE VARIOUS SHELL OF POWERSHELL

EXPECTATIONS

- MAY NOT BE BEST TOOL FOR THE JOB
- ADDING TOOLS TO THE TOOLBOX
- LIVING OFF THE LAND

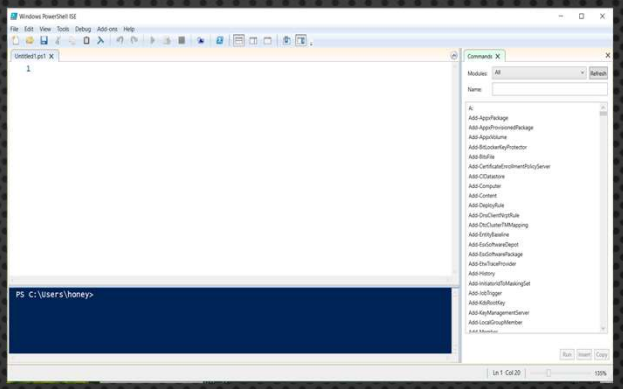
BRIEF HISTORY

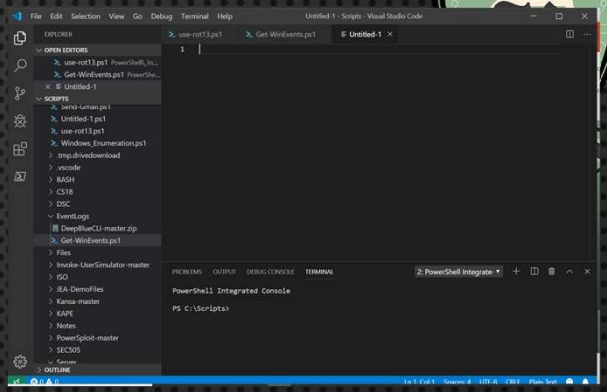
Version	Release Date	Product Shipped With
Windows PowerShell v1	2006	Exchange 2007
Windows PowerShell v2	2009	Windows Vista
Windows PowerShell v3	2012	
Windows PowerShell v4	2013	
Windows PowerShell v5	2016	Window 10
Powershell v6	2018	
Powershell v7	2020	

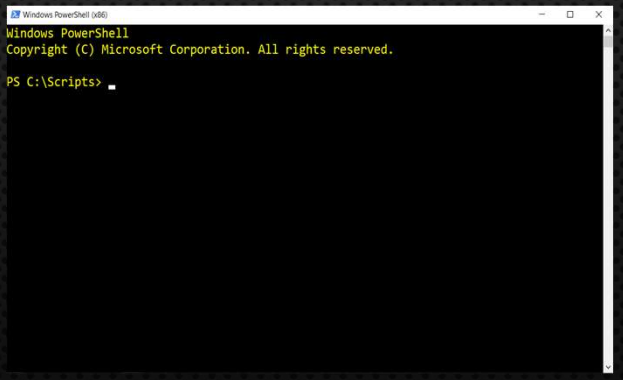
Jones, D. (2020). Shell of an Idea: The untold history of PowerShell. Lean Publishing.

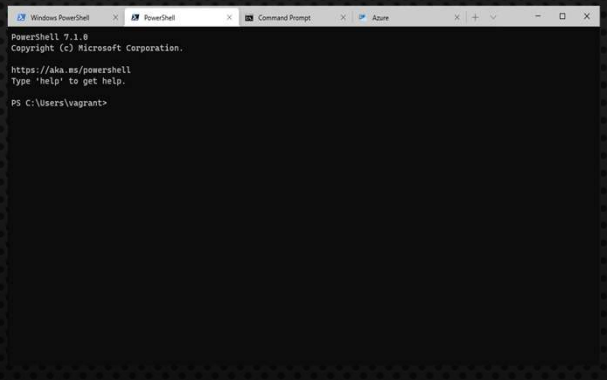
ENVIRONMENTS

- POWERSHELL CONSOLE
- POWERSHELL ISE
- VSCODE
- WINDOWS TERMINAL









CHECK ON LEARNING

- What shell does not have word wrap
  - PowerShell ISE
- What shell has word wrap, intelliscence, and can help with syntax
  - VSCode
- What shell is defaulted to a blue background
  - PowerShell
- What shell can do PowerShell, cmd, python, etc.
  - Terminal

SECURING POWERSHELL

- DISABLE POWERSHELL V2
- ENABLE TRANSCRIPT LOGGING
- ENABLE SCRIPT BLOCK LOGGING
- ENABLE MODULE LOGGING

DISABLE POWERSHELL V2

- DEFAULT ON WIN 7, WIN 8 AND EARLY WIN10
- DEPRECATED IN 2017
- POWERSHELL V5 CAN BE DOWNGRADED TO V2
- NO LOGGING

- POWERSHELL V5 IS NEW DEFAULT
- HAS TRANSCRIPT LOGGING (V3)
- HAS SCRIPT BLOCK LOGGING (V5)
- HAS EVENT LOGGING
- REMOTING CAN BE SECURED (V3)


DISABLE POWERSHELL V2

- `Get-WindowsOptionalFeature -Online -FeatureName MicrosoftWindowsPowerShellV2`
- `Disable-WindowsOptionalFeature -Online -FeatureName MicrosoftWindowsPowerShellV2Root`
- `Enable-WindowsOptionalFeature -Online -FeatureName MicrosoftWindowsPowerShellV2Root`

TRANSCRIPT HEADERS

****

PowerShell transcript start

Start time: 20220611170517

Username: 169BASEIMAGE\169user

RunAs User: 169BASEIMAGE\169user

Configuration Name:

Machine: 169BASEIMAGE (Microsoft Windows NT 10.0.22000.0)

Host Application: C:\Program Files\PowerShell\7\pwsh.dll

Process ID: 9356

PSVersion: 7.2.4

PSEdition: Core

GitCommitId: 7.2.4

OS: Microsoft Windows 10.0.22000

Platform: Win32NT

PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.10032.0, 6.0.0, 6.1.0, 6.2.0, 7.0.0, 7.1.0, 7.2.4

PSRemotingProtocolVersion: 2.3

SerializationVersion: 1.1.0.1

WSManStackVersion: 3.0

****

- **********
- Windows PowerShell Transcript Start
- Start time: 20220611171406
- Username : 169BASEIMAGE\169user
- Machine : 169BASEIMAGE (Microsoft Windows NT 10.0.22000.0)
- **********

ENABLING SCRIPTBLOCK LOGGING

- When you enable Script Block Logging, PowerShell records the content of all script blocks that it processes. Once enabled, any new PowerShell session logs this information.

- `HKLM\SOFTWARE\Wow6432Node\Policies\Microsoft\Windows\PowerShell\ScriptBlockLogging -name EnableScriptBlockLogging -value 1`

- Look in Microsoft-Windows-Powershell/Operational for event ID: 4104

https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_logging_windows?view=powershell-7.2


ENABLE MODULE LOGGING

- If you enable this policy setting and specify one or more modules, pipeline execution events for the specified modules are recorded in the Windows PowerShell log in Event Viewer.

- Import-Module <Module-Name>
- (Get-Module <Module-Name>).LogPipelineExecutionDetails = $true

- HKLM\SOFTWARE\Wow6432Node\Policies\Microsoft\Windows\PowerShell\ModuleLogging → EnableModuleLogging = 1
- HKLM\SOFTWARE\Wow6432Node\Policies\Microsoft\Windows\PowerShell\ModuleLogging\ModuleNames → * = *


FIREWALL

- Enable logging
  - set-NetFirewallProfile -Name Public -LogAllowed True -LogBlocked True -LogIgnored True
- Firewall cmdlets
  - Disable-NetFirewallRule
  - Enable-NetFirewallRule
  - Get-NetFirewallRule
  - New- NetFirewallRule
  - set- NetFirewallRule


CHECK ON LEARNING

- WHAT VERSION OF POWERSHELL NEEDS TO BE DISABLED
  - POWERSHELL V2
- WHAT CMDLET IS USED TO ENABLE ANY OF THE POWERSHELL LOGGING
  - SET-CHILDITEM


SUMMARY

- Disable PowerShell v2
- Enable Script Block logging
- Enable Transcript logging
- Enable Module logging

What did you learn in this module? What will you take back and use at work?

ELO 4 – PARSING WINDOWS LOGS

ACTION: LEARNER WILL DEMONSTRATE USING POWERSHELL TO PARSE LOGS

CONDITION: GIVEN LEARNING ACTIVITIES, READINGS, PEER AND INSTRUCTOR FEEDBACK, REFLECTION TIME, DEVELOPMENT TIME, AND PRACTICAL EXERCISES

STANDARD: GIVEN A IOC, LEARNER WILL USE POWERSHELL TO PARSE WINDOWS EVTX LOGS TO LOCATE THE IOC


GET-WINEVENT VS. GET-EVENTLOG

GET-EVENTLOG

- MICROSOFT.POWERSHELL.MANAGEMENT
- POWERSHELL 2 - 5.1
- [-NEWEST <INT32>]
- [-AFTER <DATETIME>]
- [-BEFORE <DATETIME>]
- [-MESSAGE <STRING>]
- USES A WIN32 API THAT IS DEPRECATED

GET-WINEVENT

- MICROSOFT.POWERSHELL.DIAGNOSTICS
- POWERSHELL 5.1 – CURRENT
- [-FILTERXPATH <STRING>]
- [-OLDEST]
- GETS EVENT AND CLASSIC LOGS


GET-WINEVENT VS. GET-EVENTLOG

- GET-EVENTLOG -LIST | SELECT-OBJECT LOG | MEASURE
  - COUNT 11
- GET-WINEVENT -LISTLOG * | SELECT LOGNAME | MEASURE
  - COUNT 483


XML FILTER

```
<QueryList>
    <Query Id="0" Path="Security">
        <Select Path="Security">
        *[System[(EventID=4624)]] and
        *[EventData[Data[@Name='IpAddress'] and (Data='172.16.12.3')]] and
        *[EventData[Data[@Name='IpPort'] and (Data=56842 or Data=65499 or
        Data=65497 or Data=50726)]]</Select>
    </Query>
    </QueryList>
```

CHECK ON LEARNING

- GET NUMBER OF CLEARED LOGS
- WHAT IP DID SAMIR LOGIN FROM
- WHAT WAS THE NAME OF THEIR WORKSTATION

ELO 5 SUMMARY

Action: Learner will demonstrate using PowerShell to parse logs

Condition: Given learning activities, readings, peer and instructor feedback, reflection time, development time, and practical exercises

Standard: Given a IOC, learner will use PowerShell to parse Windows evtx logs to locate the IOC


DESIRED STATE CONFIGURATION OVERVIEW

- WHAT IS DSC
- DSC COMPONENTS
- DSC CONGIGURATION FILE
- LOCAL CONFIGUATION MANAGER

DESIRED STATE CONFIGURATION (DSC)

PowerShell DSC is a configuration management platform built into Windows that is based on open standards. DSC is flexible enough to function reliably and consistently in each stage of the deployment lifecycle (development, test, pre-production, production), and during scale-out.

https://docs.microsoft.com/en-us/powershell/dsc/getting-started/wingettingstarted?view=dsc-1.1&viewFallbackFrom=dsc-2.0

DSC COMPONENTS

- Push Server
- Pull Server
- Mof Files
- Configuation Files

- Can be standalone
- Works with other configuration management tools (Ansible)


PUSH SERVER

- PUSH MODE REFERS TO A USER ACTIVELY APPLYING A CONFIGURATION TO A TARGET NODE
- START-DSCCONFIGURATION –PATH <PATH TO MOF>
- TYPICALLY USED FOR TESTING

PULL SERVER

- CLIENTS ARE CONFIGURED TO GET THEIR DESIRED STATE CONFIGURATIONS FROM A REMOTE PULL SERVICE
- CAN BE CONFIGURED AS AZURE AUTOMATION OR ONPREM SMB
- CLIENT CHECKS PULL SERVER ON A REGULAR BASIS TO LOOK FOR CHANGES
- USED FOR PRODUCTION

MANAGED OBJECT FORMAT (MOF)

- THE LANGUAGE USED TO DESCRIBE COMMON INFORMATION MODEL (CIM) CLASSES.
- THE ACTUAL FILE THAT CONFIGURES THE MACHINES
- COMPILED BY THE RESOURCE CONFIG FILE


LOCAL CONFIGURATION MANAGER (LCM)

- LCM IS THE SERVICE THAT MANAGE DSC ON THE LOCAL MACHINE
- GET-DSCCONFIGURATION
- GET-DSCLOCALCONFIGURATIONMANAGER
- REMOVE-DSCCONFIGURATIONDOCUMENT -STAGE CURRENT –VERBOSE
- SET-DSCLOCALCONFIGURATIONMANAGER -PATH 'C:\METACONFIG\LOCALHOST.META.MOF' -VERBOSE

RESOURCE CONFIG FILE

- PLAIN TEXT DOCUMENT THAT IS USED TO CREATE THE MOF
- THE RESOURCE IMPORT SECTION, THESE ARE THE RESOURCES YOU WILL BE IMPORTING
- HAS A NODE BLOCK, USED TO LIST MACHINES THAT THE CONFIG IS FOR
- THEN THERE IS THE CONFIG BLOCK, THIS HOW THE NODES WILL BE CONFIGURED


- DEMO DSC


CHECK ON LEARNING

- WHAT ARE THE PARTS OF A RESOURCE CONFIGURATION FILE?
  - RESOURCE IMPORT, NODE, CONFIGURATION

DSC SUMMARY

- WHAT IS DSC
- DSC COMPONENTS
- DSC CONGIGURATION FILE
- LOCAL CONFIGUATION MANAGER

ENABLING LEARNING OBJECTIVE / LEARNING STEP ACTIVITY 4 (ELO/LSA 4)

ACTION: LEARNER WILL DESCRIBE THE PROCESS OF USING POWERSHELL JEA

CONDITION: GIVEN LEARNING ACTIVITIES, READINGS, PEER AND INSTRUCTOR FEEDBACK, REFLECTION TIME, DEVELOPMENT TIME, AND PRACTICAL EXERCISES

STANDARD: LEARNER WILL LIST THE STEPS NEED TO CONFIGURE POWERSHELL JEA

POWERSHELL JEA

- "JUST ENOUGH ADMINISTRATION (JEA) IS A SECURITY TECHNOLOGY THAT ENABLES DELEGATED ADMINISTRATION FOR ANYTHING THAT CAN BE MANAGED WITH POWERSHELL." –MICROSOFT
- "JUST LIKE KEYS ON A KEY CHAIN" – JEFFERY SNOVER
- "REALLY COOL WAY TO EMPOWER MY USERS AND FELLOW ADMINS" – JAMES HONEYCUTT
- ALLOWS COMMON USERS TO PERFORM ADMIN FUNCTIONS (WHEN PROPERLY CONFIGURED)

HOW DOES IT WORK

- USER/ADMIN USES POWERSHELL REMOTING TO ACCESS REMOTE SERVER USING THE SESSION CONFIGURATION FILE AND USING "RUNAS"
- A VIRTUAL ADMIN ACCOUNT IS CREATED AND USED DURING THIS SESSION ONLY
- THE VIRTUAL ADMIN TECHNICALLY HAS ACCESS TO ALL CMDLETS, BUT ONLY SHOWS THE USER WHAT THEY ARE ENTITLED TO, BASED ON THE ROLE CAPABILITIES FILE
- USER DOES WHAT THEY NEED TO, EXITS REMOTE PSSESSION

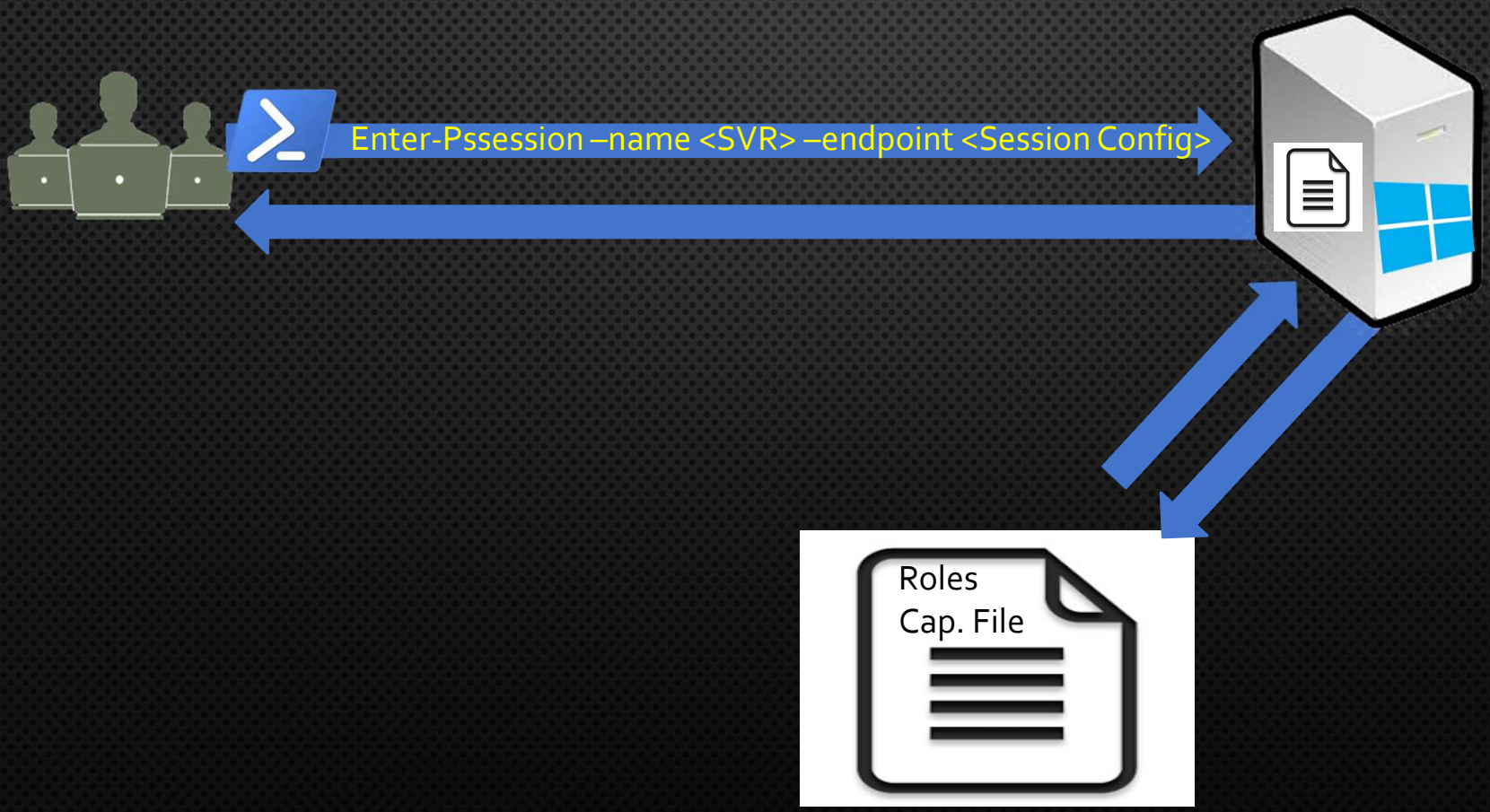
ROLE CAPABILITY FILE

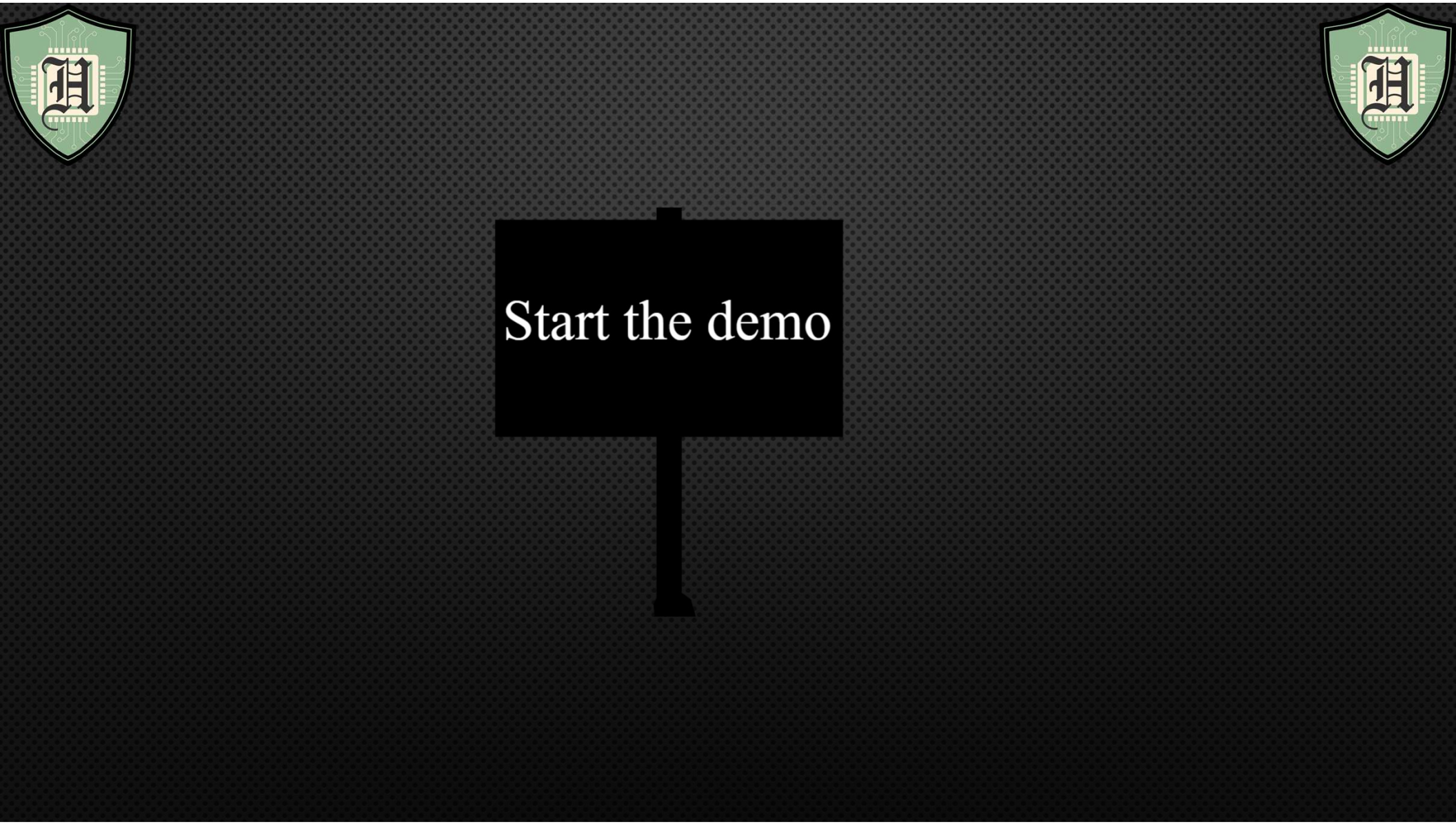
- CONTROLS WHAT CMDLETS, MODULES, FUNCTIONS, AND PARAMETERS ARE ALLOWED

SESSION CONFIGURATIONS

- CONTROLS WHO CAN LOG IN AND DETERMINES WHAT ROLE CAPABILITY FILE TO USE.
- VIRTUAL ACCOUNTS ARE CREATED ON THE FLY AND ARE USED FOR THE ONE SESSION ONLY. WILL HAVE LOCAL RIGHTS ON ENDPOINT AND MEMBER SERVERS OR DOMAIN ADMIN RIGHTS ON DC
- SPECIFIED VIRTUAL ACCOUNTS CAN BE SPECIFIED, BUT MUST BE IN THE APPROPRIATE LOCAL GROUP. GROUP MANAGED SERVICE ACCOUNTS CAN BE USED IF THE USER NEEDS NETWORK RESOURCES. (HARDER TO TRACE BACK TO A SPECIFIC USER)

HOW IT WORKS


Enter-Pssession –name <SVR> –endpoint <Session Config>
Roles
Cap. File


Start the demo

CHECK ON LEARNING

WHAT FILES DO YOU NEED TO ALLOW POWERSHELL REMOTING FROM NON-ADMIN USERS?

1. SESSION CONFIGURATIONS
2. ROLE CAPABILITIES FILE


CHECK ON LEARNING

- WHICH FILE CONTROLS THE COMMAND VISIBLE TO THE USER?
  - ROLE CAPABILITIES FILE

CHECK ON LEARNING

- WHICH FILE CONTROLS THE SESSION AND CREATES THE VIRTUAL ACCOUNTS?
  1. SESSION CONFIGURATION FILE

ELO 4 SUMMARY

ACTION: LEARNER WILL DESCRIBE THE PROCESS OF USING POWERSHELL JEA

CONDITION: GIVEN LEARNING ACTIVITIES, READINGS, PEER AND INSTRUCTOR FEEDBACK, REFLECTION TIME, DEVELOPMENT TIME, AND PRACTICAL EXERCISES

STANDARD: LEARNER WILL LIST THE STEPS NEED TO CONFIGURE POWERSHELL JEA

IS THIS SOMETHING YOU CAN START USING TODAY? IF SO HOW WILL YOU USE IT?

EVALUATION

OVERALL CHECK ON LEARNING:

REVIEW

- PowerShell Review
- PowerShell Remoting Review
- Securing Windows with PowerShell
- Hunting with PowerShell


Any
Questions