RSC
Ripple Software Consulting

Pentest +
Exploit Specifics

Author: Joseph Lee
Email: joseph@ripplesoftware.ca
Mobile: 778-725-3206

Remote Desktop Exploits

RDP Windows Remote Desktop

- Very powerful since the exploit would give attacker access to **full desktop GUI**
- Many CVE listed for recent exploits on RDP
    - https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Windows+Remote+Desktop+Protocol
- **MS RDP** uses **TCP port 3389**
- RDP may use an RDP concentrator for remote access / port forwarding
- Should use VPN / encryption for data-in-transit
- Ensure old accounts are removed
- Enforce password complexity policies

ARD Apple Remote Desktop

- Some CVE exist for ARD although none are recent
    - https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Apple+Remote+Desktop
- Known exploits have been patched
- Mac Os usually has automatic updates

VNC Virtual Network Computing

- **RFB Remote Framebuffer Protocol** is an open protocol for VNC
    - https://datatracker.ietf.org/doc/html/rfc6143
    - **RFB** works at the **frame-buffer** level exporting bitmaps generated by the video-card
- Version for Windows, Linux, MacOs
- Various products with name VNC
    - UltraVNC
    - VNC Server
    - Real VNC
- Metasploit includes VNC payloads

- Several CVEs for various products in past few years
  - https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=VNC

X-Server Forwarding

- **X11** or **X-Windows** sometimes called **X**
- GUI, desktop manager for **Linux** and **Unix** systems
- Graphical applications that support X11 such as **SSH** can have **GUI accessed remotely**

Remote Shell Exploits

Telnet

- Remote shell access
- **Not encrypted** so **very vulnerable to network sniffing**

SSH Secure Shell

- Can be configured to use **PKI Public Key Infrastructure** for secure login authentication
- Can also be configured to allow account **username and password** login
- Capturing the **first interaction** between client and server allows attacker on the same network to **potentially MiTM** since the client stores the server identity during first connection
- The first time a client connects to an SSH server the server's public key fingerprint can ensure you are connecting to the correct server
- Extracting contents of **RAM** can reveal struct used in SSH data encryption such as key, iv, cipher algorithm

rsh / rlogin / rexec

- **rsh / rshd –** Remote shell
  - Uses **TCP port 514**
- **rlogin –** Remote login
- **rexec –** Remote execution
- Legacy system for Unix / Linux
- Unix system files
  - /etc/hosts/equiv
  - /home/$USER/.rhosts
- Make for good target because **data is not encrypted**

Physical Penetration

Pretexting

- Presenting a **fictional scenario** to members of the organization
- **Data collected** using OSINT, watering-hole attacks, or other information gathering campaigns can be **used to create a pretext**
- Important to keep **contact information on-hand** in case of unexpected events
    - Who to contact in the organization when something goes wrong
    - Plan for dealing with unexpected encounters with facility staff
    - What to do if you end up in jail or otherwise detained

Information gathering

- **Dumpster diving**
- Even **shredded documents** can be put back together
- **OSINT**, search for maps other important documents allow knowledge of the physical landscape before entering
- **Social engineering** can cause employees or ex-employees to disclose information
- **Manufacturer and model** of locks, security cameras, card-scanners, or other devices can lead to **generating specific attacks** against those devices

Entering facilities

- **Egress sensors**
    - Automatically open doors
    - Can be used to access
    - Should be mapped
    - Some can be manipulated with magnets
    - Jamming a specific radio frequency may render inoperable
- **Fencing**
    - May include camera security, guards, motion detection, barbed-wire, razor-wire, lighting, etc
- **Lockpicking**
    - Know the laws on carrying lock-picking tools in your jurisdiction
    - Keep certifications on-hand
    - Locks on doors, desks, filing cabinets may be opened using simple tools
    - The Open Organization of Lock-pickers
        - https://toool.us/
    - Wide variety of other door penetration tools such as **shove keys, shims, other basic materials**

    - More advanced lockpicking technology includes cameras that can be inserted into keyway to capture images of pins
- **Piggybacking & Tailgait**
    - Mantraps can prevent multiple people from entering at the same time

Social Engineering

- https://capec.mitre.org/data/definitions/403.html
- https://capec.mitre.org/data/definitions/416.html
- **SET Social Engineering Toolkit –** Built into Kali Linux
- **Psychological Aspects to Social Engineering**
    - https://capec.mitre.org/data/definitions/427.html
    - Leveraging cognitive and social psychology to cause someone to disclose information or perform an action on behalf of the attacker
    - **Trust**
        - Attacker is able to present the victim with some type of information or other social influence that will gain the victims trust
        - This could include items such as company letterhead, uniform, or other items
    - **Pretexting / Impersonation**
        - https://capec.mitre.org/data/definitions/407.html
        - Creates an invented scenario, assuming an identity or role to persuade a targeted victim to release information or perform some action
    - **Reciprocation / Quid Pro Quo**
        - https://capec.mitre.org/data/definitions/418.html
        - Attacker does something to elicit reciprocation from the attacker maybe by holding a door open for them
    - **Authority**
        - https://capec.mitre.org/data/definitions/421.html
        - Conveying a sense of authority that motivates the target to reveal specific information or take specific action
    - **Urgency / Scarcity**
        - https://capec.mitre.org/data/definitions/420.html
        - Conveying a perception of scarcity, or a situation of limited supply, the adversary aims to create a sense of urgency in the context of a target's decision-making process
    - **Social proof / Likeliness / Similarity**
        - https://capec.mitre.org/data/definitions/424.html
        - Leveraging the inherent human nature to assume behavior of others is appropriate
        - Attacker leverages the victims need to feel included similar to attacker
    - **Elicitation**

        - https://capec.mitre.org/data/definitions/410.html
        - Using any combination of social engineering methods to gain information
    - **Interrogation**
        - https://capec.mitre.org/data/definitions/434.html
    - **Bribery**
        - Paying the victim or giving them something else in order to change their behaviour to disclose information
    - **Phishing attacks**
        - https://capec.mitre.org/data/definitions/98.html
        - Sending email to a large group of victims to deceive them into clicking links, download documents or software attachments
    - **Vishing**
        - https://capec.mitre.org/data/definitions/656.html
        - Calling a victim to attempt to persuade or deceive them to disclose information, or perform some action on behalf of the attacker
    - **SMS Phishing**
        - https://capec.mitre.org/data/definitions/164.html
        - Targets mobile phone users with a phishing attack
    - **Whaling**
        - Attempting a phishing attack that is specially crafted to deceive an executive level person
    - **Spear phishing**
        - https://capec.mitre.org/data/definitions/163.html
        - A specially crafted and targeted phishing attack aimed at a narrow group of people such as single organizational department
- **Shoulder surfing**
    - https://capec.mitre.org/data/definitions/508.html
- **Watering hole attacks**
    - By visiting websites or physical locations where people from an organization are known to visit, intel can be gathered through direct communication with members of the organization
- **Cloned websites / Pharming**
    - https://capec.mitre.org/data/definitions/89.html
    - https://capec.mitre.org/data/definitions/543.html
    - By creating a fake website with similar appearing domain users can be tricked into entering sensitive information such as login credentials, or download malicious files or software
- **USB key drops**
    - Leaving USB keys where other people will find then and insert them into computers
    - Mailing USB keys with malicious files on them to an organization and attempting to get them to access the files
    - Physical honeypots

    - Also called **baiting**
    - Labelling or other ways to make the bait attractive

Software Exploitation

Injection Vulnerabilities

- **Input Validation**
    - **Input Whitelisting**
        - Define the specific input **parameters that are allowed** into the application
        - Checking that **input matches the expected** input type, range, etc.
        - **Best method** of validating input when possible
    - **Input Blacklisting**
        - Define the specific input that **is not allowed** into the application
        - **Not as good as whitelisting** input
        - Example is to **remove <script> tags**, or SQL commands from user input
    - **Input encoding**
        - **Escaping characters** that can cause problems such as during database insertion
        - One example is to escape single-quotes in database insertion
- **Parameter Pollution**
    - https://capec.mitre.org/data/definitions/460.html
    - **Insertion multiple fields** with the same key into data such as GET variables
    - Hoping that the application will **mishandle** them, or **not sanitize** the second value
- **Code Injection**
    - https://capec.mitre.org/data/definitions/242.html
    - Supply some code to application, web application or service
    - Can be used to **DOS the target** or attain **code execution** through target
- **Code Inclusion**
    - https://capec.mitre.org/data/definitions/175.html
    - Forcing arbitrary code to be retrieved locally or from a remote location and executed
    - **Differs from code injection** in that code injection involves the direct inclusion of code while code inclusion involves the addition or replacement of a reference to a code file
    - **Remote Code Inclusion**
        - https://capec.mitre.org/data/definitions/253.html
    - **Local Code Inclusion**
        - https://capec.mitre.org/data/definitions/251.html
- **Command Injection**

  - https://capec.mitre.org/data/definitions/248.html
  - When some input to the application is evaluated as commands and executes by a subprocess in the application
  - Injects new items into an existing command thus modifying interpretation away from what was intended
  - **OS Command Injection**
    - https://capec.mitre.org/data/definitions/88.html
    - Multi-stage attack
      - Inject operating system commands into existing application functions
      - Issue specially crafted input into the application to trigger the use of the injected commands
- **Format string attack**
  - https://owasp.org/www-community/attacks/Format_string_attack
  - Occurs when the submitted data of an input string is **evaluated as a command** by the application
  - **Format Function** is an **ANSI C** conversion function, like **printf, fprintf**, which converts a **primitive variable** of the programming language into a **human-readable string representation**
  - **Format String** is the argument of the **Format Function and is an ASCII Z string** which contains text and format parameters, like: **printf ("The magic number is: %d\n", 1911);**
  - **Format String Parameter**, like **%x %s** defines the type of conversion of the format function
  - Attack could be executed when the application doesn't properly **validate the submitted input**

Source Code Comments

- Source code comments can **reveal information** about the application's functionality
- Allows attackers easier ability to understand the code
- May reference how **credentials are encrypted / unencrypted** which would allow attacker to decrypt out-of-band
- Comments should be stripped out of the application before compilation or use on the production server

Error Handling

- Wrong error handling can give attacker access if **fail-open security check** is used
- Verbose error handling can reveal the internal functions of the application
  - https://cwe.mitre.org/data/definitions/209.html
- **Error verbosity settings** may allow errors to be printed to screen which should not be such as **SQL query strings**, or **file not found**

Environment Variables

- Modification of environment variables can allow an attacker to change the expected behaviour of a program
  - https://capec.mitre.org/data/definitions/13.html
- Environment variables values may contain credentials such as API authentication keys
- Buffer Overflow via setting environment variables with large values that will cause the application to write to restricted memory space
  - https://capec.mitre.org/data/definitions/10.html

Hard-Coded Credentials

- **Access to source code** in a server breach can reveal and **hard-coded credentials** such as 3rd party API secret keys
- **Environment variables** can be used such that the credentials are not available in pain-text
- Access to environment variables through command such as **printenv** and **env** can be removed from the production server
- Credentials can be encrypted and decrypted at runtime, but the encrypted credentials can be exfiltrated and decrypted since the method is included in the source-code
- Even credentials only stored in RAM can be found but are much harder access

Race Conditions

- https://capec.mitre.org/data/definitions/26.html
- Vulnerability can be triggered when the security of a code segment depends on the sequence of events within the system
- **TOCTTOU Time of Check to Time of Use** occurs when permissions are checked to far in advance in a situation when access may be revoked by the time the resource is requested
  - https://capec.mitre.org/data/definitions/29.html
- Spectre and Meltdown exploited a race condition in speculative execution where code could be pushed onto the CPU stack before file permissions have been checked
- Race conditions can be leveraged by using **symbolics links**
  - https://capec.mitre.org/data/definitions/27.html

Code Signing Attack

- https://capec.mitre.org/data/definitions/206.html
- **Code signing** allows the end-user to **validate that the software package has been issued by the original developer**
- Attacker extracts credentials used for code signing from a production environment and then uses these credentials to sign malicious content with the developer's key

Unsigned Code

- **Subverting code signing**
    - https://capec.mitre.org/data/definitions/68.html
- If the code is unsigned, it could be from a malicious 3rd party
- https://capec.mitre.org/data/definitions/477.html

Application Security Testing

- **SAST Static Application Security Testing**
    - Static code analysis of source code can be done by an **automated tools** or by **manual inspection**
    - The program is not run, but rather inspected for potential vulnerabilities
    - https://owasp.org/www-community/Source_Code_Analysis_Tools
    - **FindBugs** and **findsecbugs**
    - **SonarQube**
    - **YASCA Yet Another Source Code Analyzer**
- **DAST Dynamic Application Security Testing**
    - Relies on the execution of code to find bugs
    - Can be done via automated tools or manually
    - Can be used to test new features are working properly
    - **Interception Proxies**
        - Used to modify data being sent between the application and a server
        - Used for fuzzing
        - Can be browser extension or stand alone application
        - Proxy software
            - **Firefox Tamper Data**
            - **ZAP Zed Attack Proxy**
            - **Burp Suite**
- **Fuzzing**
    - Allow proxies to **alter the input being sent** to web-application or other application / service hoping to crash the application or gain shell access
        - https://capec.mitre.org/data/definitions/28.html
    - Allow input to be sequentially sent programatically to **test for error handling** malformed input
        - https://capec.mitre.org/data/definitions/215.html
- **Decompilation**
    - https://capec.mitre.org/data/definitions/190.html
    - Compiled applications can be **decompiled to reveal the source code**
    - Original variable names have been replaced with enumerated variable name placeholders
    - Attacker can to **modify and recompile a malicious** version of the application

    - Attacker can **search for bugs / errors** in the code that can trigger vulnerability such as remote code execution
- **Debugging**
    - Debugging allows line by line execution of an application to inspect the state of memory address during the application's process
    - Can be used to **inspect the state of the underlying system** during fuzzing or other code injection testing
    - In Windows adding a user debug privileges to an application gives them admin like privileges

Exploiting Host Vulnerabilities

Authentication Vulnerabilities

- **Accounts Traversal**
    - Usually initial exploitation leads to access to an account with limited privileges
    - Escalating privileges relies on further gathering information
        - UserID
        - Hashed passwords
        - Poorly secreted services and software
        - Default configurations
        - Other attacks
- **Default Account Settings**
    - Devices / Systems / applications with default passwords not changed by admin during installation
    - Lists of device default usernames and passwords can help find
        - https://cirt.net/passwords
        - Service or device accounts with admin access privileges may be easier

Remote Access

- **SSH**
    - Using stolen credentials to access system via SSH
- **NETCAT and Ncat**
    - Setup a reverse shell to issue commands to target shell
- **Proxies and Proxychains**
    - Used to setup a remote access path to a target that is unavailable directly from the initial end-point
- **Metasploit remote access**
    - Some Metasploit plug-ins can gain remote shell access and return that shell access to msf console

Virtual Machines

- https://capec.mitre.org/data/definitions/480.html
- Can **check network interfaces** to see if system / services are virtualized
- **wmic baseboard get manufacturer, product** on Windows to
- **system-detect-virt** on systems running **systemd**
- **demidecode** can provide similar information
- **ls -l /dev/disk/by-id** can list attached storage devices to check if they are virtualized
- **Detecting hypervisor** can help look for **CVEs**
  - **VMWare**
  - **Xen Project**
  - **Hyper-V**
  - **VirtualBox**

Container Attacks

- Compromising the application that runs in the container
- **No So Secure** provides VM with vulnerable **Docker container**
  - https://notsosecure.com/vulnerable-docker-vm/

Credential Attacks

- **Credential Acquisition**
  - **Mimikatz** has ability to read hashes and passwords directly from memory
  - Kali Linux has 3 tools for exfiltrating Windows passwords as part of **creddump**
    - https://tools.kali.org/password-attacks/creddump
    - **cachedump** dumps cached passwords
    - **lsadump** passwords **LSA Secrets**
    - **pwdump** dumps password hashes
  - Linux passwords stored in **/etc/shadow**
  - **LaZagne** simple python script can search Mac, Linux, and Windows for exposed credentials
  - **Other tools** can search specific OSs for exposed credential files
- **Offline Password Cracking**
  - Passwords can be **brute-forced** offline
    - https://capec.mitre.org/data/definitions/49.html
  - **Rainbow cracking**
    - https://capec.mitre.org/data/definitions/55.html
    - **Hash to cleartext files** that make it easy to find password when hash present in the rainbow table
  - **Dictionary attacks**

- **Lists of common passwords** or **passwords obtained through data-breach** that can be served into password cracking software to find a match
- **Specific Tools**
  - **Hashcat**
    - Password cracking utility that uses GPUs
    - Faster than other tools
  - **John the Ripper**
    - The go-to password cracking utility for pentesters for long time
    - Wide range of functions
  - **Cain and Able**
    - Designed to work with Windows NT, XP, and 2000
  - **Hydra**
    - AKA **thc-hydra** a brute-force dictionary attack tool
    - Can be used against SSH, http/https, SMB, and databases
    - hydra -l [userid] -p [wordlist] [target ip] -t [timing] [protocol]
  - **Medusa**
    - Similar to Hydra
    - https://foofus.net/goons/jmk/medusa/medusa.html
  - **Patator**
    - More difficult to use than Hydra or Medusa
  - **Wordlists and Dictionaries**
    - **CeWL Custom Wordlist Generator**
      - Spiders a website looking for Keywords that can be used in dictionary attack
    - **W3AF Web-application and Attack Audit Framework**
      - Web-application framework security scanner that includes directory and filename brute-forcing
    - **DirBuster**
      - Older software last updated in 2013
      - Some useful Java application designed to brute-force directories

- **KeyLoggers**
  - **Physical devices** can be **place in between keyboard USB and USB port**
  - **Software** can be installed that can **capture keystrokes**

Creating Persistence

- **Scheduled Tasks and Cron**
  - Useful to maintain persistent access
  - Defenders should monitor these systems for rouge injected commands
  - Can survive reboots of the system
  - The process will not always be actively running / sleeping to avoid detection

    - **Windows:**
        - SchTasks /create /SC Daily /TN "Calculator" /TR "C:/Windows/System32/calc.exe" /ST 08:00
    - **Linux:**
        - Cron has many directories with cron files for each user and system
            - /etc/cron.hourly
            - /etc/cron.daily
            - /etc/cron.weekly
            - /etc/cron.monthly
        - **echo "* * * * * /path/to/script.sh" crontab -e**
- **Add rouge service**
    - Services are started each time the system boots
    - The process will be always running / sleeping
- **Shimming an Existing Application / Service**
    - Replace an existing piece of software with a trojan version
    - The software will appear as normal process to user
    - The user will start the software for normal use and therefore maintain persistence

Exploit Chaining / Combination Attacks

- Involves **multiple exploits** or attacks that are **chained together** to fully compromise a device
- Most individual vulnerabilities do not immediately give the level of compromise desired by attacker
- **Analysis of the end-point architecture** and **associated vulnerabilities** may uncover possible chain of vulnerabilities which can lead to desired level of compromise

Linux Hosts

- Harder to exploit than Windows due to open source nature of the OS
- Less attacked than Windows due to fewer users
- **SETUID/SETGID – SETUID Set User ID and SETGID Set Group ID**
    - https://attack.mitre.org/techniques/T1548/001/
    - **Permission bits on executable files** that determine **who can run the file**
    - Needed for tasks that require different privileges than what the user is normally granted
    - Any users able to execute a file with SUID or SGUID will automatically execute with the privileges of the file's owner (commonly root) and/or the file's group
    - **Finding files with SETUID set**
        - Shows all SETUID files and folders
            - **find / -perm -4000**
        - Finds all SETUID files

      - **find / -perm -u=s -type f 2>/dev/null**
    - Get SUID files with details
      - **find / -user root -perm -4000 -exec ls -ldb {} \;**
  - **Setting SETUID SETGUID**
    - To set the permission bit on a file
      - **chmod u+s <filename>** or **chmod g+s <filename>**
    - To remove the permission bit on a file
      - **chmod u-s <filename>** or **chmod g-s <filename>**
  - **shebang**
    - **#!** at the start of the script allow setting the shell with with to execute the commands in the script (Ex: **#! /bin/bash**) is known as **shebang**
    - Some older Linux / Unix systems **allow SETUID and SETGUID scripts to be run** when the shebang is set
      - http://www.faqs.org/faqs/unix-faq/faq/part4/section-7.html
  - Kernels are configured to **prevent SETUID scripts** from working
  - Some executables with **SETUID** can allow privilege escalation such as: (Nmap, Vim, find, Bash , More, Less, Nano, cp)
  - **Sticky bits**
    - Also known as **restricted delete permissions** flags
    - The tmp directory uses sticky bits to determine ownership of files there
    - **ls -l | grep tmp**
- **Insecure SUDO**
  - sudo allows users to **escalate permissions to root**
  - Affected by **SELINUX** when enabled
  - List user with sudo permission
    - **/etc/sudoers /etc/sudoers.d**
    - **sudo -l**
- **Restricted shells**
  - Prevent users from changing directories, setting **PATH** or **SHELL** variables
  - When confronted with restricted shells:
    - Check commands you can run, looking for **SUID** commands
    - Check if you can use **sudo**
    - Check for languages like Python, Pearl, or Ruby
    - Download compiled C executables
    - Try redirect operators such as | , >, and escape characters
    - **rbash, bash -r, rksh, ksh -r, rsh, sh -r**
- **Ret2libc – Return to libc**
  - **Buffer overflow attacks** that target the **C library**
  - Modern system that use **ASLR address system layout randomization** help prevent such buffer overflow attacks
  - Advanced attacks can try to circumvent **ASLR**

- **Linux Kernel Exploits**
    - Linux kernel exploits are listed and described in CVE's
        - https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Linux+Kernel
    - Code execution, privilege escalation, bypass attacks are most useful
    - Some practice available in Metasploitable
    - Vulnerabilities maybe patched with updates
    - Check operating system release
        - **lsb_release -a**
        - **uname -a**

Windows Hosts

The **majority of corporate workstations** are Windows

Windows System Exploits

- **RPC Remote Procedure Call**
    - The **RPC is a concept** that can be implemented in different ways
    - It is not a standard, rather similar to the API concept
    - Modern attacks often have **RPC** exploits available
- **DCOM Distributed Component Object Model**
    - Proprietary Microsoft software component
    - Similar to **RPC**
    - Allows COM objects to communicate with each other over the network
    - Common way to attack **Windows NT, 2000, NT, and 2003 Server**
- **PsExec**
    - **Sysinternals** Windows toolkit includes **PsExec**
    - Allows admins to run programs on **SMB** port **445**
    - Vector allows to **run arbitrary commands remotely**
    - Good vector if found enabled during a pentest
    - Most **malware will flag PsExec** commands if detected against a server
    - Exploits available in Metasploit
    - https://toshellandback.com/2017/02/11/psexec/

- **PS Remoting / WinRM Windows Remote Management System**
    - **WinRM** enables **PowerShell** and **supports remote** PowerShell
    - WinRM runs as a windows service
    - Remote PowerShell is not turned on by default
    - To turn on **Remote PowerShell** (admin account required)
        - **enable-PSRemoting -force**
        - If systems are not on same domain, setup trust between domains
        - **Set-Item wsman:\localhost\client\trustedhosts [ipaddress or**

hostname]
    - Restart **WinRM** required
- **WMI Windows Management Instrumentation**
  - Allows for remote management and data gathering
  - Installed on all Windows systems
  - Can allow remote command execution, file transfers, and registry data
  - Provides:
    - **Windows Defender information** to **SNMP**
    - Application Inventory
  - Many exploits available:
    - **WMImplant**
      - https://github.com/FortyNorthSecurity/WMImplant
      - **PowerShell based** tool that uses WMI to attack targeted machines
      - Also **includes C&C** for issuing commands and receiving results
      - Functions for **lateral movement** through network
      - **basic_info** looks for logged in users
      - **vacant_system**
    - **WmiSploit**
      - https://github.com/secabstraction/WmiSploit
      - PowerShell scripts that leverage the WMI service, for post-exploitation use

Obtaining Credentials

- **cPassword**
  - Passwords used to be stored in **cPassword** attribute in **Windows Group Policy** items
  - Accessible to any authenticated user
  - Microsoft published the **AES encryption key** used to encrypt passwords in **cPassword**
  - Cracking **cPasswords** can be done with Metasploit **post/windows/gather/credentials/gpp**
  - Also available via **PowerSploit** modules: **Get-ChachedGPPPassword, GetCPPPassword**
  - **$SYSVOL** file named **Groups.xml**
  - Microsoft offered fixes **MS14-025**
  - https://docs.microsoft.com/en-ca/archive/blogs/ash/dont-set-or-save-passwords-using-group-policy-preferences
  - https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/store-passwords-using-reversible-encryption
- **Cleartext Passwords in LDAP**
  - **LDAP Lightweight Directory Access Protocol built** into **AD Active Domain**
  - Used for many authentication services in **AD**

  - Commonly misconfigured
  - **AD** does not force **SSL/TLS**
  - **LDAP Simple Binds** will expose credentials by sending them in plaintext
  - To check if LDAP signing is not enforced check the **Directory Service Logs** for even IDs **2886** and **2887** which reports how many cleartext binds occurred in last 24 hours.
- **Kerberoasting**
  - https://github.com/nidem/kerberoast
  - Relies on requesting **service tickets** for **service account service principle names (SPNS)**
  - Tickets are encrypted with the password of the service account associated with SPN
  - PowerSploit **Get-NetUser** or **Powershell** can be used to gather list of accounts
  - Request service tickets via Powershelll
  - **Mimikatz** extracts the service ticket data **kerberos::list/export** command
  - Crack using offline cracking tools
  - **kirbi2john.py** is a tool to crack kerberos password hashes
  - **Four Steps to Kerberoasting**
  - **Scan AD** for user accounts with **service principle names (SPN)** set
  - Request service tickets using SPN
  - Extract service tickets from memory and save to file
  - Conduct offline brute force attack against passwords in service tickets
- **NTLM hash**
  - A service account can be used to create forged Kerberos service ticket called **Silver Ticket** using **Mimikatz**
  - **NTLM hashes** are unsalted so they can be replayed across Windows authentication services
  - **Pass-the-hash** injecting hashes into **LSASS**
  - **Pass-the-hash** into **SMB, WMI**
  - **Sysinternals psexec** tool can directly accept an NTLM hash as authentication instead of password
- **Credentials in LSASS Local Security Authority Subsystem Service**
  - LSASS enforces security policies on security systems
  - LSASS on **Windows 7 / Windows Server 2008** stored passwords in cleartext
  - Can be extracted using **Mimikatz** and other tools
  - **Windows 8 / 10, Windows Server 2012 / 2016** encrypt passwords
  - However, registry settings **Wdigest authentication** can be changed to allow cached credentials to access passwords in cleartext
    - **HKEY_LOCAL_MACHINE/Security/Policy/Secrets**
  - **Impacket** and **Metaspliot** have modules to attack LSASS
- **LSA Secrets**
  - **LSA Secrets** registry location contains encrypted passwords for logged in users
    - **HKEY_LOCAL_MACHINE/Security/Policy/Secrets**

    - Encryption key is stored in **Parent Policy Key** in the **Windows registry**
    - Admin access to the registry allows recovery of encrypted password and key
- **Unattended Installation**
    - An **.xml** can be used to perform an unattended installation of Windows
    - **Windows Deployment Services (WDS)** encodes the local admin password in **plaintext or Base-64**
    - Credentials can be found in multiple locations after an unattended installation
    - Metasploit module **post/windows/gather/enum_unattend** can retrieve unattended passwords
    - **Can be found in following locations**
        - C:\unattend.xml
        - C:\Windows\Panther\unattend.xml
        - C:\Windows\Panther\Unattend\unattend.xml
        - C:\Windows\system32\sysprep.inf
        - C:\Windows\system32\sysprep.xml
    - **SAM Windows Security Account Manager Database**
        - Contains password hashes
        - With appropriate privileges **Mimikatz** can dump the password hashes
- **Stored Credentials**
    - 3rd party software will store credentials on the system that can be plaintext, or replayed
    - Examples include
        - **Putty** stores credentials in cleartext in the Windows Registry
            - **HKCU/Software/SimonTatham/Putty/Sessions**
        - **McAffee** password for endpoint protection software stored in **SiteList.xml** file
        - **UltraVNC** stores passwords in **ultravnc.ini** file, located in the same folder as **winvnc.exe**
- **Keyloggers**
    - https://capec.mitre.org/data/definitions/568.html
    - Many proprietary and free keyloggers available
    - Not very difficult to build a custom keylogger
    - Metasploit metaterpreter keylogger **keyscan_start** and contents viewed with **keyscan_dump**
    - Metasploit can capture Windows login credentials
        - Get process ID of **winlogin.exe**
        - **migrate** command can migrate keylogger to that PID
- **Windows Credential Manager**
    - Stores various passwords like browser passwords and network resource passwords
    - Elevated privileges allows a full dump of the passwords stored within
    - Using **LaZagne** with elevated privileges can retrieve passwords in plaintext

DLL Dynamic Link Library Hijacking / Injection

- https://pentestlab.blog/2017/04/04/dll-injection/
- DLLs are **software modules** that can be accessed / shared by applications and services in Windows
- Extensions include: **.dll, .ocx, .cpl, .drv**
- **DLL hijacking** replaces existing DLL loaded by applications
- **DLL search order hijacking**
  - https://attack.mitre.org/techniques/T1574/001/
  - Applications loading DLLs and the resources follow a search order hierarchy
    1. The current application working directory
    2. Windows system directory
    3. All **$PATH** directories
  - **Write permission** for the current working directory of an application can allow **rouge DLLs** to be injected
- **Changing the Registry entries for known DLLs**
  - The registry can be modified to **change the expected location of a system DLL** to point to a **rouge DLL**
- **Side-loading DLLs**
  - https://attack.mitre.org/techniques/T1574/002/
  - Side-by-side function in Windows activates when **multiple copies of the same DLL are required**
  - Application requires a manifest that lists the correct DLL
  - DLLs are loaded in to **C:\Windows\WinSxS**
- **Phantom DLLs**
  - https://attack.mitre.org/docs/training-cti/Cybereason%20Cobalt%20Kitty%20-%20original%20report.pdf
  - Some DLLs have been depreciated, but Windows will still has the DLL included
  - These **legacy DLL's** may include vulnerabilities and can be loaded by an application

Unquoted Service Paths

- https://attack.mitre.org/techniques/T1574/009/
- When windows starts a service, it looks for executable location
- All service paths should be quoted ""
- How it unquoted service paths can be exploited
  - Windows has a **search hierarchy** when supplied with a path
    - C:\Program.exe
    - C:\Program Files.exe
    - C:\Program Files\Unquoted.exe
    - C:\Program Files\Unquoted Path.exe
    - C:\Program Files\Unquoted Path Service.exe
    - C:\Program Files\Unquoted Path Service\

  - If the service path is not quoted and there are spaces in the path, the system will **look for executable files** in the directory with the same name as the first part of the path
  - **Write permissions** to one of the directories in the path can allow attacker to upload a **malicous.exe** file with the same name as the first segment of the path that includes spaces
  - Files will be executed at the permission level of the SYSTEM
- Finding unquoted service paths
  - **wmic** can be used to find unquoted service paths
    - **wmic service get name,displayname,pathname,startmode |findstr /i /v “C:\Windows\\” | findstr /i /v “””**
  - Metasploit module **/exploit/windows/local/trusted_service_path**

Writeable Services

- https://attack.mitre.org/techniques/T1574/010/
- If the directory a service executable is in allows write permissions the executable file can be replace with a rouge version of the same name
- How to find
  - **SysInternals acesschk** can be used to find directories that the user (or service user) has write permissions to
  - Metasploit **/exploit/windows/local/service_permissions**
  - PowerSploit **Get-ModifiableService** and **Invoke-serviceAbuse**

Windows Kernel Exploits

- https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Windows+kernel
- Metaspliot **post/windows/gather/enum_patches** lists any missing patches
- Metasploit also has other kernel exploit modules
- Kernel flaws have been found in every version of Windows and Windows Server OS
- Most require local access

Insecure File / Folder Permissions

- https://cwe.mitre.org/data/definitions/732.html
- Overly broad file permissions on files or folders
- System administrators often loosen file restrictions to make it easier
- **AccessEnum** and **AccessChk**
- Powershell **Get-Acl**
  - Documentation: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.security/get-acl?view=powershell-7.1
  - Gets the security descriptor for a resource, such as a file or registry key

- **icacls**
  - Documentation: https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/icacls
  - Used to check and manage permissions on Windows systems
  - **cacls.exe** is predecessor to **icacls.exe**
  - **icacls.exe /?** provides help on the command
  - Examples: https://www.computerhope.com/icacls.htm

Physical Device Security

Cold-boot attacks

- https://en.wikipedia.org/wiki/Cold_boot_attack
- **Remove RAM**
  - Removing memory modules and placing in to the attackers system
  - Used to **capture encryption keys** from RAM without having to circumvent memory restrictions / permissions
  - Using **cooling agent** can increase time allowed to swap volatile memory
- **Boot From USB Drive**
  - Using a **bootable USB drive to boo**t
  - Can use forensics software to make full image or copy select info from system hard-drive
  - Use **digital forensics tool-kits** to ignore file permissions on the slaved system drive
  - Light-weight operating system can preserve some of the RAM data from before reboot, which can be scraped for encryption keys or other data

Serial Consoles

- RJ45 connection or serial 9-pin
- Serial console connection is not dependant on network connectivity (Layer 2) since it's connected directly to the appliance
- Used as **last-resort** access to devices
- Require **local physical access** to the console or RJ45 cable to tap, splice
- There are certain benefits to using a serial console:
  - The console provides a **secure, physical, and dedicated access**
  - Network connectivity issues cannot interrupt this type of connection, and management traffic cannot be intercepted
  - It is secure because of its direct connection
  - When configuring over a serial port, you are not using any type of network connectivity

    - To change Internet Protocol (IP) addressing on the firewall, using the serial console is an excellent option

JTAG Debug Pins and Ports

- Industry standard for hardware debug ports
- Named after the **Joint Test Action Group**
- https://hackaday.com/2016/12/15/the-many-faces-of-jtag/
- https://www.pentestpartners.com/security-blog/the-art-of-finding-jtag-on-pcbs/

Scripting and Penetration Testing

URL / ASCII / Percent Encoding

- Space = %20
- ! = %21
- " = %22
- # = %23
- $ = %24
- % = %25
- & = %26
- ` = %27
- (= %28
-) = %29
- * = %2A
- + = %2B
- , = %2C
- - = %2D
- . = %2E
- / = %2F

Bash Bourne Again Shell

- Common on **Linux and Mac Os**
- Can be command line or run from executable scripts
- **#!/bin/bash** (shebang) at top of file tells OS to use bash
- Bash is also the default on modern Linux and MacOs
- **chmod u+x** hello.sh gives file executable permissions
- **echo "Hello World"** used to print to terminal
- **.sh** is common file extension
- **numbers=(1,2,3)** to create array **${numbers[0]}** to select from array
- **-eq, -lt, -le, -gt, -ge, -ne** are comparison operators

- No explicit string concatenation ($string1$string2)
- **if** condition **then** commands, **elif** condition **then** commands, **else** commands **fi**
- **for** variable **in** range **do** commands **done**
- **while** condition **do** commands **done**
- Does not have built-in functions to handle errors

PowerShell

- Developed by Microsoft and can be installed on Linux and MacOs but generally only used on Windows
- Comes preinstalled on Windows machines
- **.ps1**is common file extension
- Five policies configured using **Set-ExecutionPolicy [policy]**
    - **Restricted** blocks all use of PowerShell
    - **AllSigned** requires that scripts be signed by trusted publisher
    - **RemoteSigned** allows local scripts, but downloaded scripts must be signed
    - **Unrestricted** allows all scripts to be run buy prompts to confirm downloaded scripts before running
    - **Bypass** allows all scripts and does not warn about downloaded scripts
- **Write-Host "Hello World"** prints to terminal
- **$numbers = 1,2,3** to create array **$ages[0]** to select from array
- **-eq, -lt, -le, -gt, -ge, -ne** are comparison operators
- **+** used for string concatenation
- **if (**condition**) {**commands**} elseif (**condition**) {**commands**} else {**commands**}**
- **for (start, test, increment) {** commands **}**
- **Do {** commands **} While(**condition**)**
- **Try {} catch {}** used for error handling

Python

- General purpose **interpreted** programming language
- Can also subprocess to use system commands
- **print ("Hello World")** prints to the terminal
- **numbers = [1,2,3]** to create array **numbers[0]** to select from array
- **==, <, <=, >, >=, !=** are comparison operators
- **+** used for string concatenation
- **if/elif/else** uses **:** and end of condition statement and code block must be indented
- **for** variable **in** range**:** and code block of commands must be indented
- **while** condition**:** and code block of commands must be indented
- **try: except:** used for error handling

Ruby

- General purpose **interpreted** programming language
- Can also subprocess to use system commands
- **puts "Hello World"** prints to the terminal
- **numbers = [1,2,3]** to create array **numbers[0]** to select from array
- **==, <, <=, >, >=, !=** are comparison operators
- **+** used for string concatenation
- **if/elsif/else** condition uses **end** at the end
- **for** variable **in** range **do** commands **end**
- **while** condition commands **end**
- **begin rescue end** used for error handling

Web Applications

Good start to an external pentest contract with goal of gaining deeper access

WAF Web Application Firewall

- Can be an appliance in the network DMZ, or a preprocessing application
- Pre-processing / sanitizing the input before it reaches the web-application

Directory Traversal

- https://capec.mitre.org/data/definitions/126.html
- A request for a resource can be malformed to attempt to escape from the application root directory and access resources on the system outside of the intended application scope
- Can be used to **GET** or **PUT** malicious files onto the server
- Traversal is bound by the permissions to access that resource
    - Can be defined by server configuration files such as **httpd.conf**
    - Can be defined by file / directory permissions on the server

Session Attacks

- **Session hijacking**
    - https://capec.mitre.org/data/definitions/593.html
    - MiTM can examine HTTP headers to **extract cookies** from the communication between user and an **unencrypted website**
    - **Cookies** are used for **statefulness** (logged in status) on a website
    - Malware can also be used to **extract cookies from HTTPS headers**
    - Attacker can replay these cookies to authenticate to the website impersonating the victim's session

  - Possibly done blindly from another IP while spoofing original src IP and user-agent to trick server
  - MiTM can issue requests to the web server **impersonating the victim's** session
  - Application should compare request parameters such as the **source IP address** and **user-agent** to mitigate

Invalidated Redirections

- Some web-applications use a URL in the GET request to redirect the user’s browser
- A target can be supplied with a URL containing a redirection can be used to redirect a user to a malicious website
- Redirects in the URL should be validated to contain same origin domain or other whitelisting

Insecure Direct Object Reference

- https://capec.mitre.org/data/definitions/76.html
- When an ID or other variable points to a resource, this ID can be enumerate to attempt to access other resources.
- Example is an ID number for a user such as **id=5**
- User can simply change the id number in the url
- If the server does not check authorization of the session to access that resource, attacker can gain access to unauthorized resources
- Web-application need to **validate that the authenticated user** has access to the resource **before it is served**

File Inclusion

- **Local file Inclusion**
  - https://capec.mitre.org/data/definitions/252.html
  - Attacker can upload a file to a web-application and if that file is executed can deliver payload or attempt to exploit the system
- **Remote File Inclusion**
  - https://capec.mitre.org/data/definitions/193.html
  - Attacker can reference a remote file which can be executed

XSS Cross Site Scripting

- https://capec.mitre.org/data/definitions/63.html
- A means to inject arbitrary JavaScript into an webpage
- The JavaScript will run in the victims browser
- Attempts to forward information to the attacker, or to exploit the browser's JavaScript Engine directly (browser escape)

- **Reflected XSS**
  - https://capec.mitre.org/data/definitions/591.html
  - Includes **<script>** tags in a url which could be included in the DOM when the request serves the HTML
- **Stored / persistent XSS**
  - https://capec.mitre.org/data/definitions/592.html
  - When **<script>** is submitted buy attacker and stored in a database and served into page requests in places such as user comments, etc.
- **DOM based XSS**
  - https://capec.mitre.org/data/definitions/588.html
  - Malicious script is inserted into the HTML being parsed by a web browser
  - DOM-based XSS attack executes sometime after the page loads
  - Example is a **malicious browser extension that will inject JS script** into all pages, or selected pages

CSRF / XSRF Cross Site Request Forgery

- https://capec.mitre.org/data/definitions/62.html
- An attacker making a request to another domain, different than the original domain
- Extent of abilities depends on the user being logged into the other site or not
- Embedded Javascript can allow an attacker to conduct this type of attack
- For example, a person visiting domain.com may click on a link that would try to change their password on another site
- Some sites may use this to make another site look like it receives more traffic than it really does

Clickjacking

- https://capec.mitre.org/data/definitions/103.html
- Links that lead to malicious content
- Links that modify local browser configuration

SQL Injections

- https://capec.mitre.org/data/definitions/66.html
- **Query Parameterization**
  - A parameterized query (also known as a prepared statement) is a means of pre-compiling a SQL statement so that all you need to supply are the column names and values
  - SQL Injection is best prevented through the use of parameterized queries
  - Most languages come with modules that do parameterized queries (PDO for PHP for example)
- **Stored Procedures**

    - A stored procedure (also termed proc, storp, sproc, StoPro, StoredProc, StoreProc, sp, or SP) is a subroutine available to applications that access a relational database management system (RDBMS)
    - Such procedures are stored in the database data dictionary
    - Uses for stored procedures include data-validation (integrated into the database) or access-control mechanisms
- **Content-based Blind SQL Injection**
    - https://capec.mitre.org/data/definitions/7.html
    - Checking whether the application is vulnerable to SQL injection by testing if the application will interpret injected code
    - If the application is processing injected code, further exploitation can take place
- **Timing Based Blind SQL Injection**
    - Uses the amount of time the application takes to process a request to determine if the application is vulnerable to SQL injection similar to content-based blind injection
    - Includes a **WAIT FOR DELAY '00:00:15'; —** with a function that parses the results of the query
    - In some cases can determine the letters of the string retrieved by the database

Wireless Exploits

- **WEP**
- **WPA / WPA2**
- **WPA3**
- **Evil Twins / Rouge AP**
    - **Aircrack-ng / Airbase-ng**
        - Can be used to create evil twin attack
        - Other tools include **Aircrack-ng**, **Kismet**, **Wifite**
        - https://www.kismetwireless.net/
        - https://www.aircrack-ng.org/
        - Relies on the rouge AP being closer, or higher strength signal than the original AP broadcast signal
- **Karma attacks:**
    - https://insights.sei.cmu.edu/blog/instant-karma-might-still-get-you/
    - Listens to probe request for networks and impersonates them
- **Z-wave protocol**
    - S2 downgrade attack to S0
- **Fragmentation attacks against WEP**
    - Much harder to sniff credentials / other data since most traffic is encrypted in modern network communication
- **WPS**
    - Get from Security+ Notes
    - Pixie dust attacks

- **Bluetooth**
  - Not enough vulnerabilities found
  - **Bluejacking**
    - Get from Security+ Notes
  - **Bluesnarfing**
    - Get from Security+ Notes
  - **BlueBourne**
    - Allows attacker to take complete control over the device
    - Fetch network information
    - Execute remote code
    - Pivot to other bluetooth devices
- **Other Wireless Protocols and Systems**
  - https://hackaday.com/tag/hackrf/
  - https://www.rtl-sdr.com/black-hat-software-defined-radio-talks/
- **RFID Cloning**
  - https://capec.mitre.org/data/definitions/399.html
  - Access cards, ID cards, similar tokens
    - Low frequency – 1225-134.2 kHz
    - High frequency – 13.56 MHz (same as NFC)
    - Ultra-high frequency – 865-928 MHz
- **Jamming**
  - https://capec.mitre.org/data/definitions/604.html
  - Not a common technique
  - Filling the target WiFi spectrum frequencies with noise
  - Can disable systems that rely on WiFi communication
  - May help to avoid detection in Wifi Camera's, Alarms, IoT, etc.
  - Jamming is **not legal** in all jurisdictions
    - https://www.fcc.gov/general/jammer-enforcement

Other Protocol Exploits

SMB Server Message Block

- https://www.hackingarticles.in/impacket-guide-smb-msrpc/
- https://book.hacktricks.xyz/pentesting/pentesting-smb
- https://pentestlab.blog/2017/12/13/smb-share-scf-file-attacks/
- File sharing protocol for Windows
- Linux uses Samba which is SMB compatible
- Predecessor for Windows is **CIFS Common Internet File System**
- SMB 2 or 3 is common now
- SMB provides name resolution, file services, authentication, authorization, print

services

- **Kali** includes **SMB Scanner** to detect OS that SMB service is running on since this determines the exploit that can be used
- **Metasploit** also has SMB scanning capabilities such as **brute-force login** and enumerating **SMB services**
- **Responder** tool can acquire credentials for SMB services and get hashed credentials to crack

Kerberos

- **Administrator account attacks**
  - Attacker gains access to the admin account and can change configuration or authorization to resources
- **Ticket reuse**
  - Despite Kerberos's complex authorization granting process, tickets can be replayed if sniffed from the network traffic
- **Ticket Granting Tickets**
  - Ticket granting tickets are called **Golden Tickets** since they can be used to grant persistent access to resources
- **Kerberoasting**
  - Attacks on the kerberos authentication hashes

IPSec VPN

- **ike-scan**
  - Source Code: https://github.com/royhills/ike-scan
  - Manual: https://linux.die.net/man/1/ike-scan
  - Command-line tool that uses the IKE protocol to discover, fingerprint and test IPsec VPN servers